

Дифференцируемая графика

Part 2: Shading and Neural Volumetric Rendering

Николай Патакин, Дмитрий Сенюшкин — AIRI

Ранее в сериале

Ранее мы рассмотрели общий пайплайн дифференцируемой растеризации, используя предположение, что цвет пикселя определяется текстурой, заданной на поверхности меша.

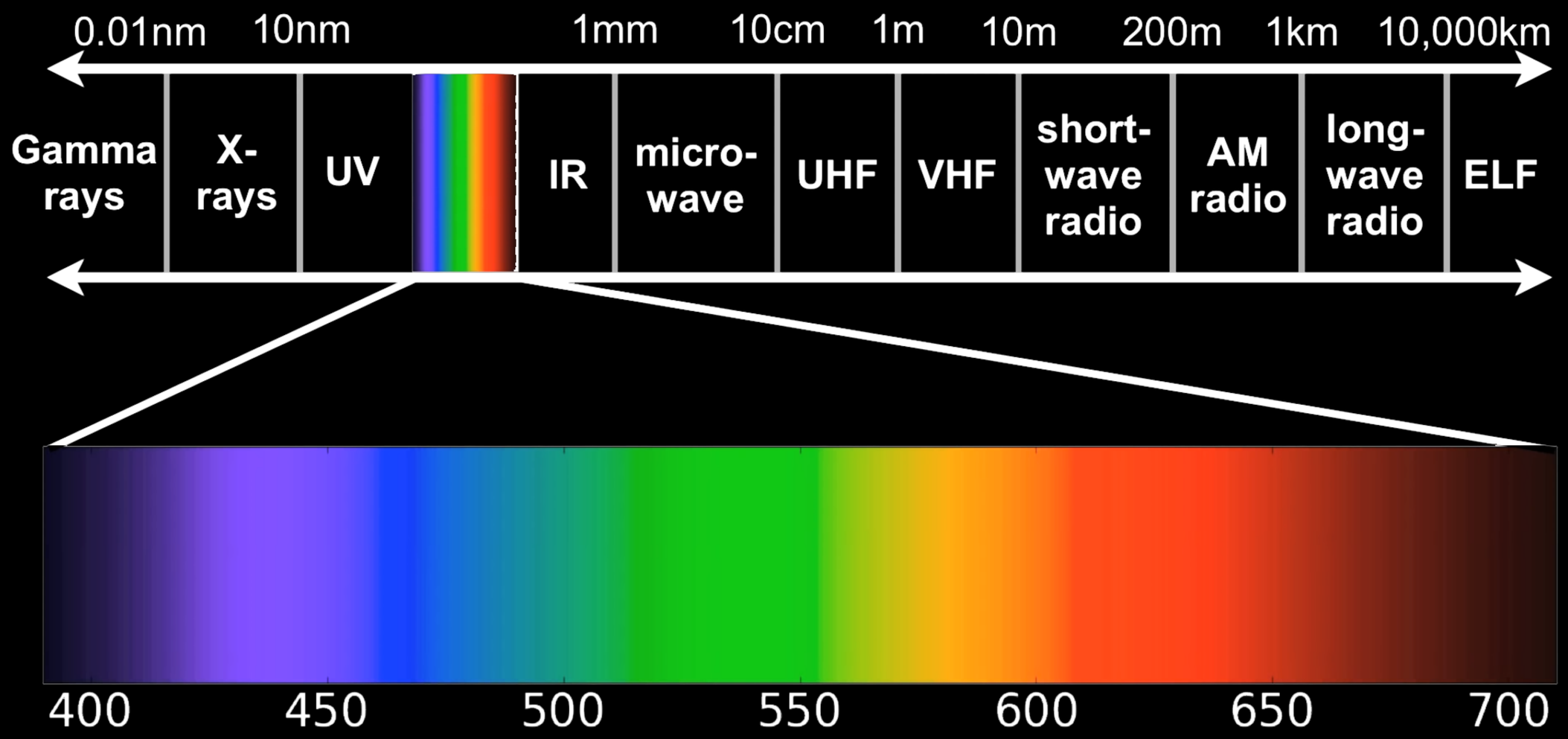
Но верно ли это предположение?

Нет! Необходимо уметь моделировать отражения и прочие view-dependent эффекты!

Этап моделирования взаимодействия света и поверхности — SHADING



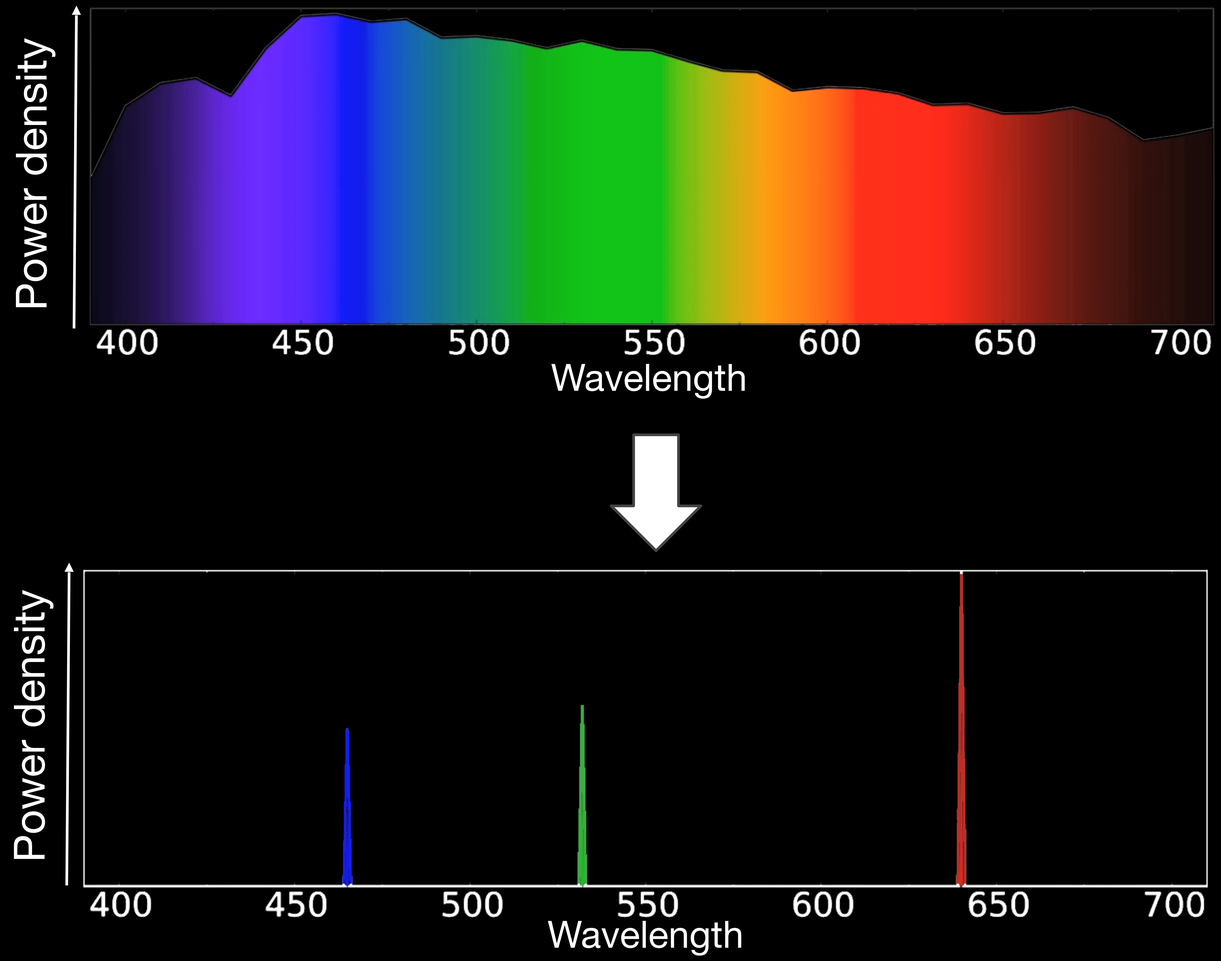
Свет и цвет


0.01nm
10nm
1mm
10cm
1m
10m
200m
1km
10,000km
Gamma rays
X-rays
UV
IR
micro-wave
UHF
VHF
short-wave radio
AM radio
long-wave radio
ELF
400
450
500
550
600
650
700

Свет и цвет

Человеческий глаз неспособен различить полноценный спектр и спектр света, состоящий из волн трех длин

Аппроксимация:
Spectral power density – смесь трех дельта функций



Нелинейность восприятия

Линейный градиент с точки зрения восприятия:



Нелинейность восприятия

Линейный градиент с точки зрения восприятия:



Физически линейные интенсивности, соответствующие ему:



Зрение человека гораздо лучше воспринимает
темные оттенки!

Свет и цвет

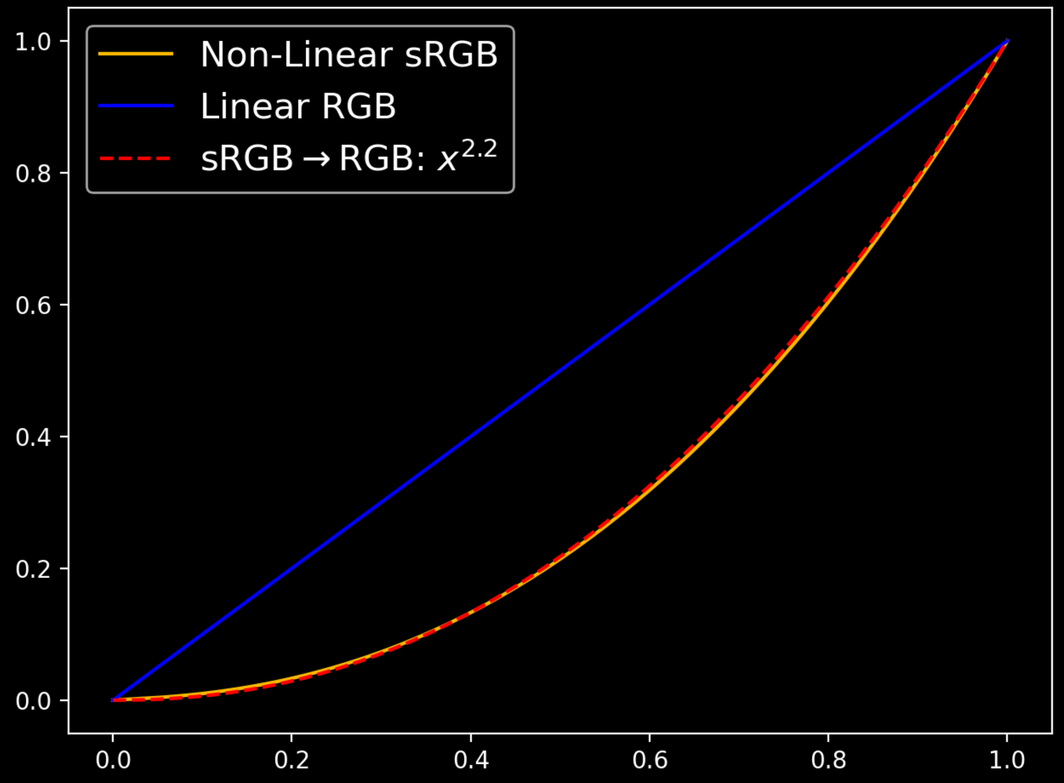
Linear RGB vs Non-linear sRGB

Для эффективного кодирования цвета (gamma correction):

$$c_{rgb} \approx c_{sRGB}^{2.2}$$

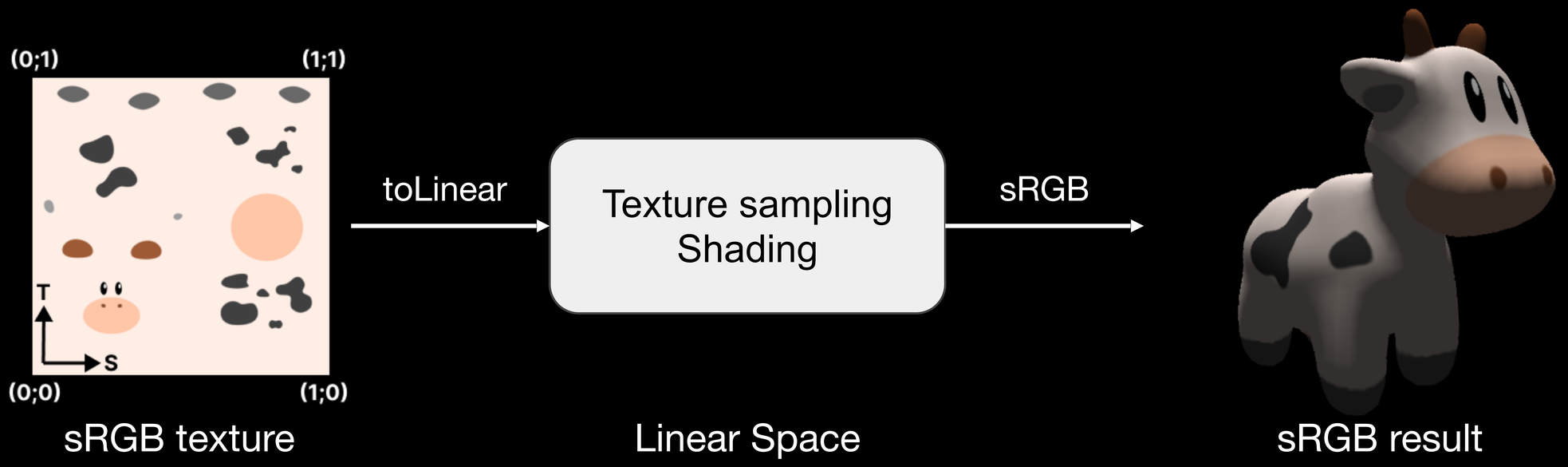
$$c_{sRGB} \approx c_{rgb}^{1/2.2}$$

Все операции с цветом должны происходить в линейном пространстве!



Свет и цвет

Корректный пайплайн шейдинга работает в линейном цветовом пространстве:



Свет и цвет

Пример: Интерполяция красного и зеленого цвета

sRGB -> Interpolate:



Проблема: Промежуточные оттенки темнее

sRGB-> Linear -> Interpolate -> sRGB:





Surface normals



World position

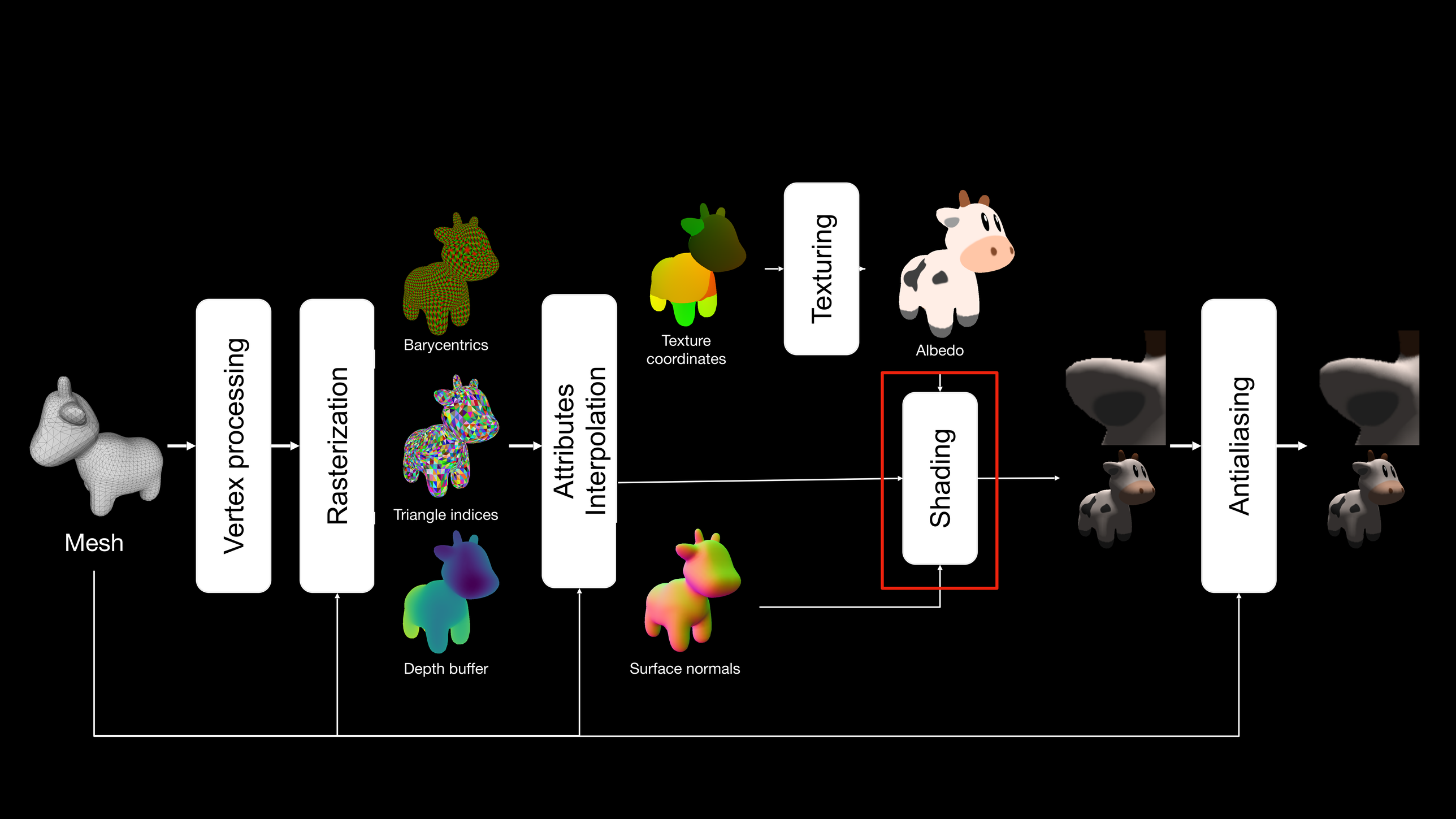


Sampled texture



Shaded image

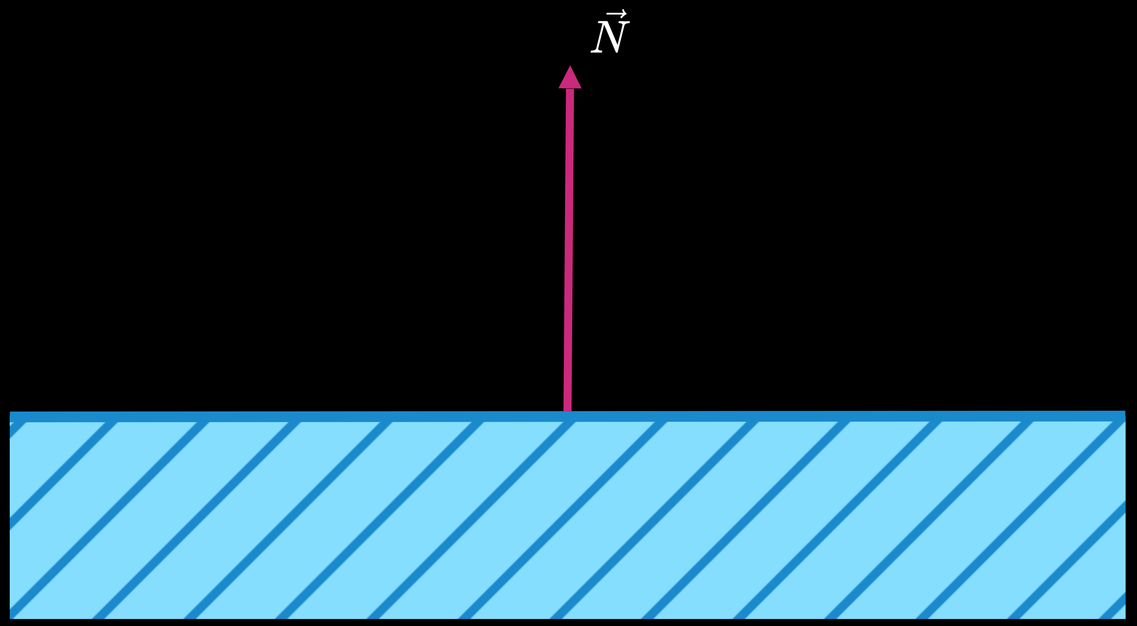
Shading


Mesh
Vertex processing
Rasterization
Barycentrics
Triangle indices
Depth buffer
Attributes Interpolation
Texture coordinates
Texturing
Albedo
Surface normals
Shading
Antialiasing

Расчет освещения

Определения

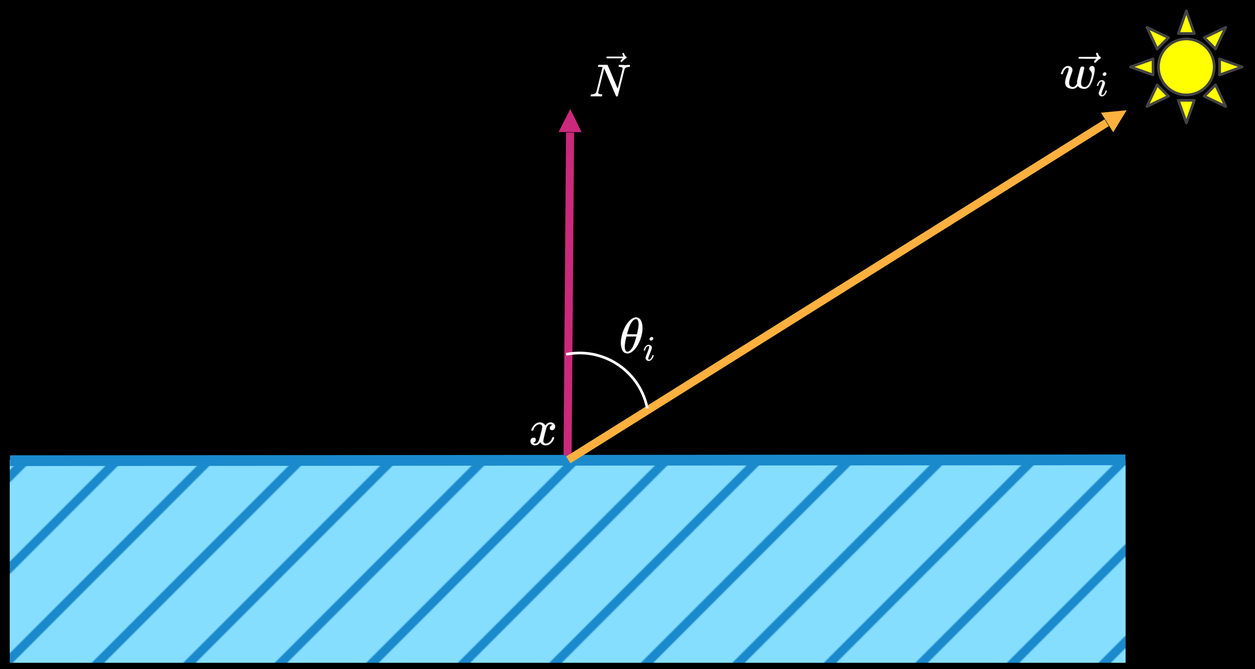
$\vec{N}$ – нормаль поверхности



Расчет освещения

Определения

$$\cos\theta_i = \langle \vec{N}, \vec{w}_i \rangle$$



$\vec{N}$ – нормаль поверхности

x – точка поверхности

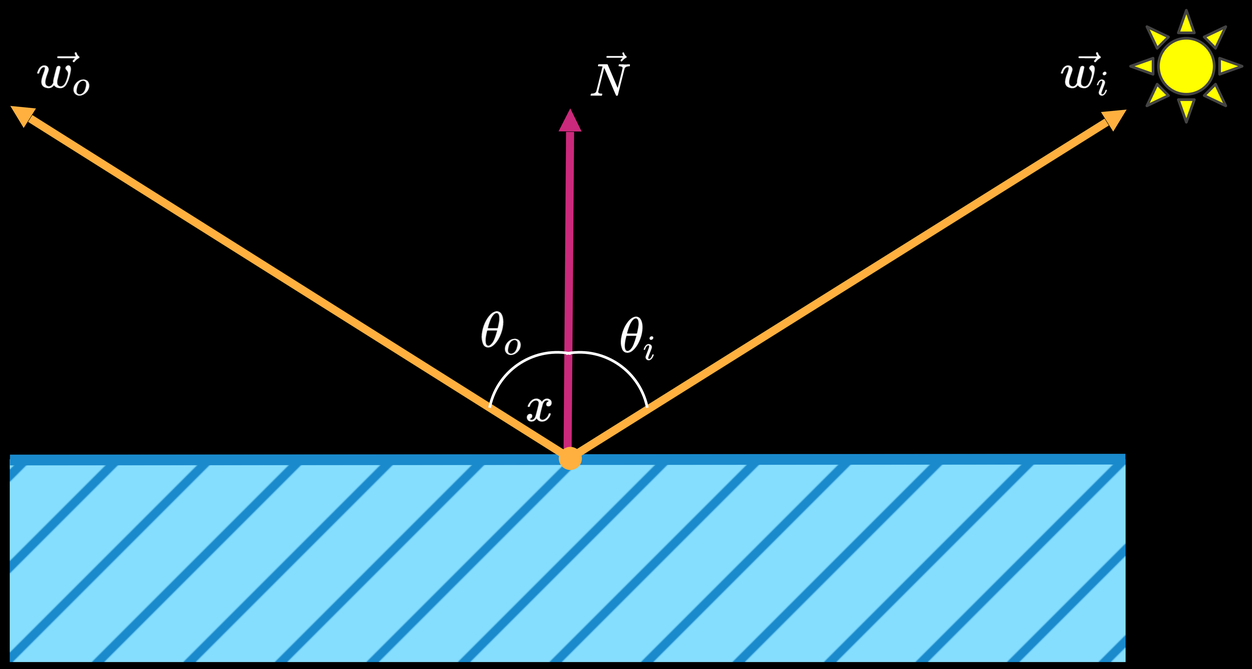
$\vec{w}_i$ – направление входящего света

θ_i – угол падения света

Расчет освещения

Определения

Идеальное отражение: $\theta_i = \theta_o$



$\vec{N}$ – нормаль поверхности

x – точка поверхности

$\vec{w}_i$ – направление входящего света

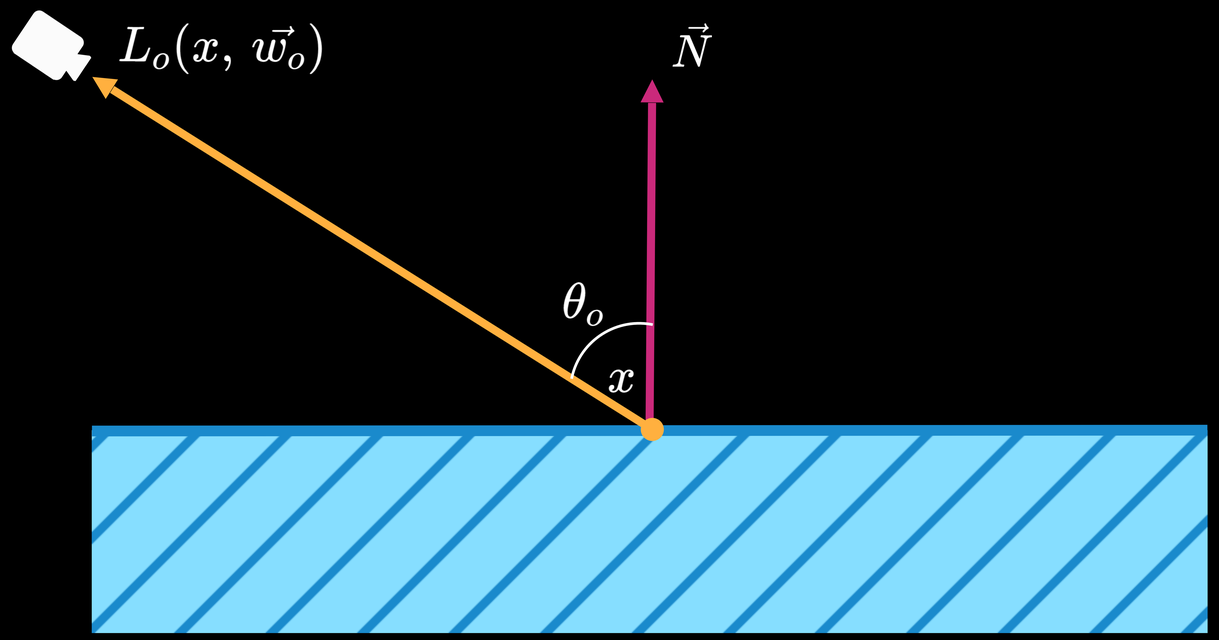
θ_i – угол падения света

$\vec{w}_o$ – направление отражения

θ_o – угол отражения света

Расчет освещения

Rendering equation

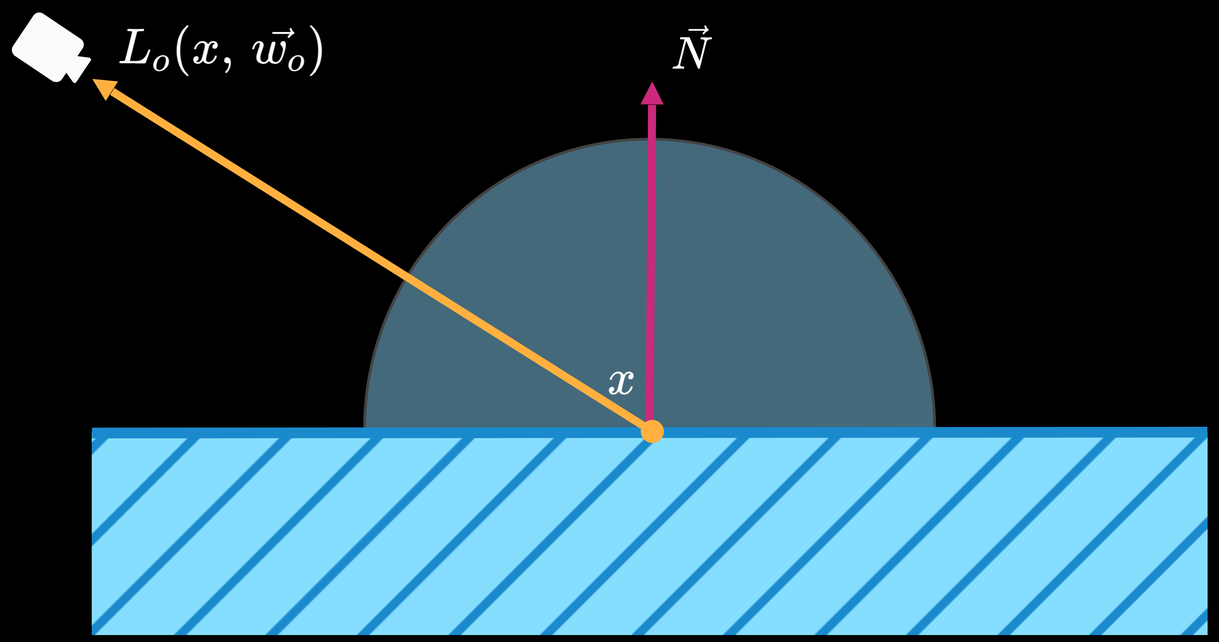


$$L_o(x, \vec{w}_o) = ?$$

Расчет освещения

Rendering equation

$$L_o(x, w_o) = \int_\Omega L(x, w_i) \cdot ? \cdot dw_i$$

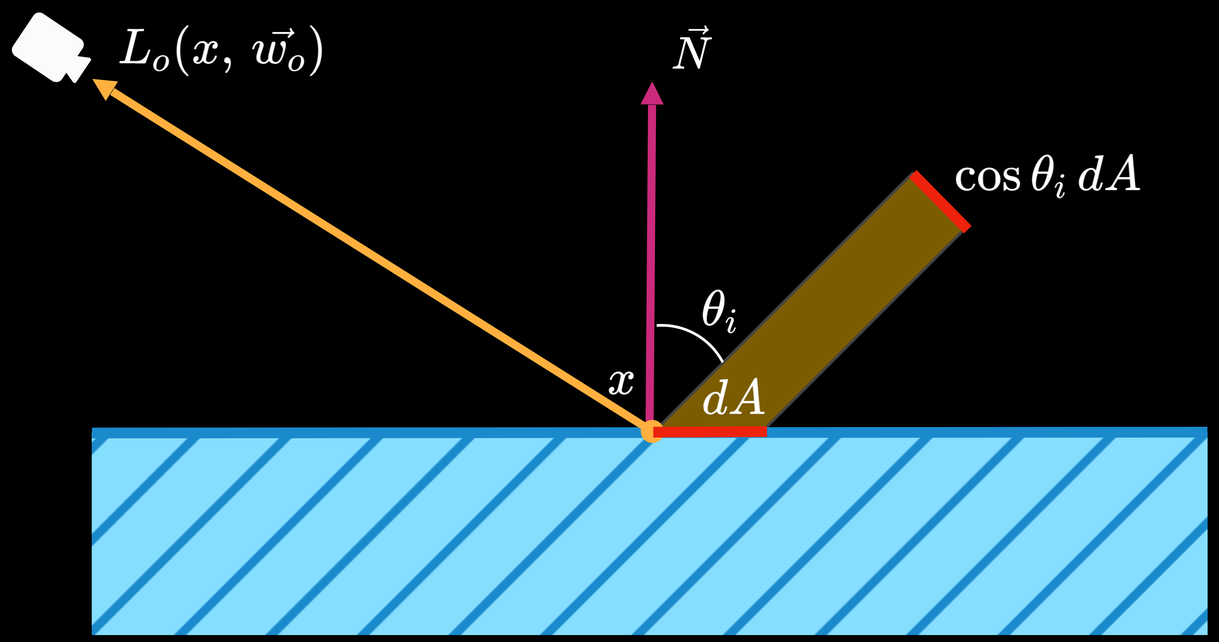


1. Свет приходит со всей полусферы Ω

Расчет освещения

Rendering equation

$$L_o(x, w_o) = \int_{\Omega} ? \cdot L(x, w_i) \cos\theta_i \, dw_i$$



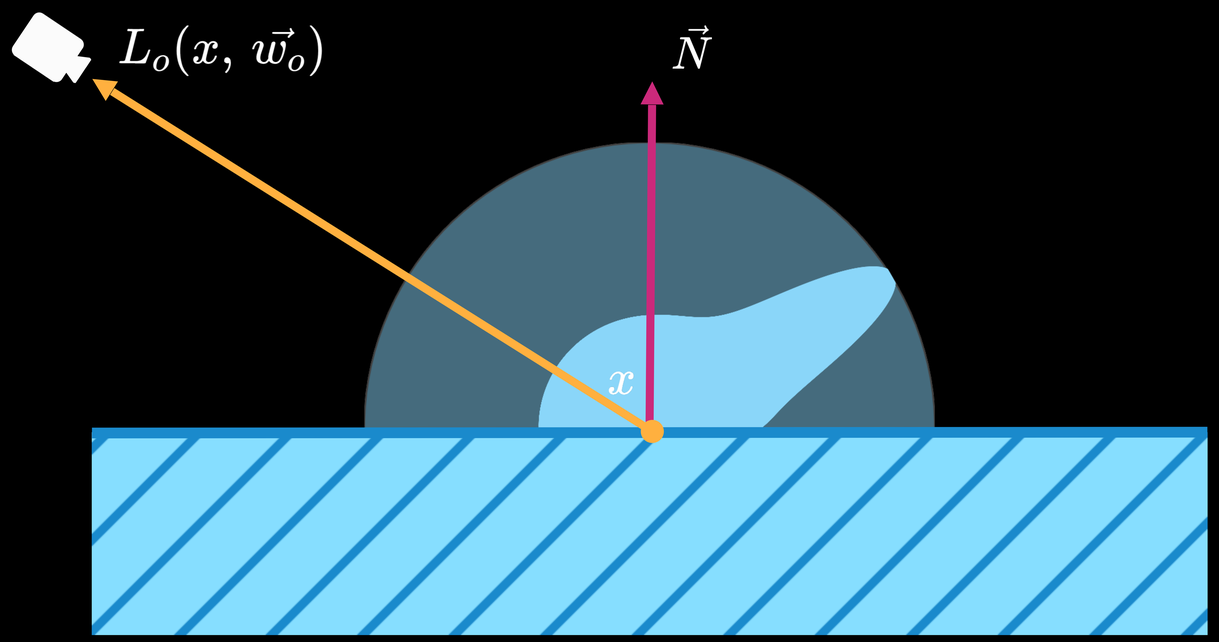
1. Свет приходит со всей полусферы Ω
2. Чем ближе источник света к линии горизонта, тем меньший вклад он имеет

Зависимость описывается косинусом угла падения света

Расчет освещения

Rendering equation

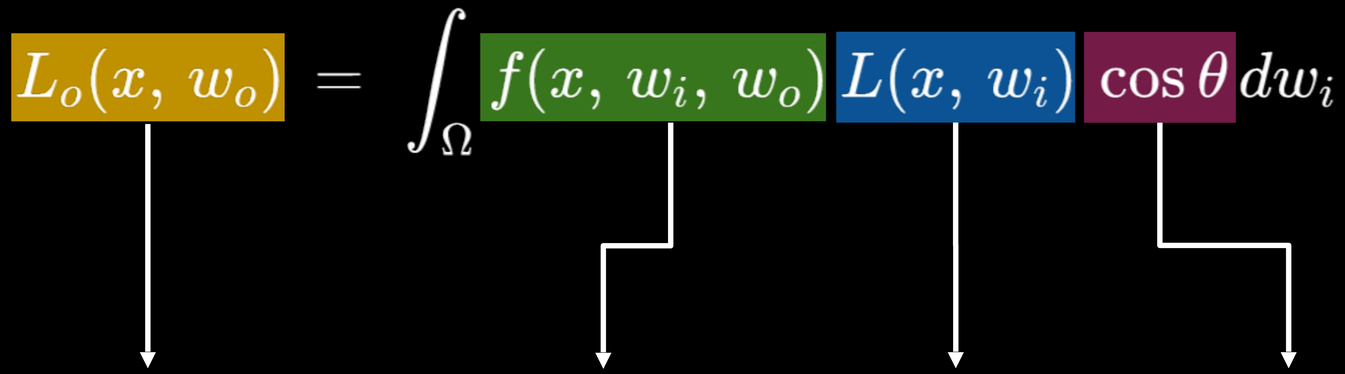
$$L_o(x,\, w_o) \;=\; \int_{\Omega} f(x, w_i, w_o) L(x,\, w_i) \cos\theta_i \; dw_i$$



1. Свет приходит со всей полусферы Ω
2. Чем ближе источник света к линии горизонта, тем меньший вклад он имеет
   Зависимость описывается
   косинусом угла падения света
3. Разные материалы отражают входящий свет в различных пропорциях

Расчет освещения

Rendering equation



Расчет освещения

Свойства BRDF

1. Неотрицательность:

$$f(x, w_i, w_o) \geq 0$$

Входящий свет не может оказывать отрицательное воздействие на исходящий

2. Принцип обратимости Гельмгольца:

$$f(x, w_i, w_o) = f(x, w_o, w_i)$$

Взаимная замена сенсора и источника света не влияет на результат измерения

3. Закон сохранения энергии

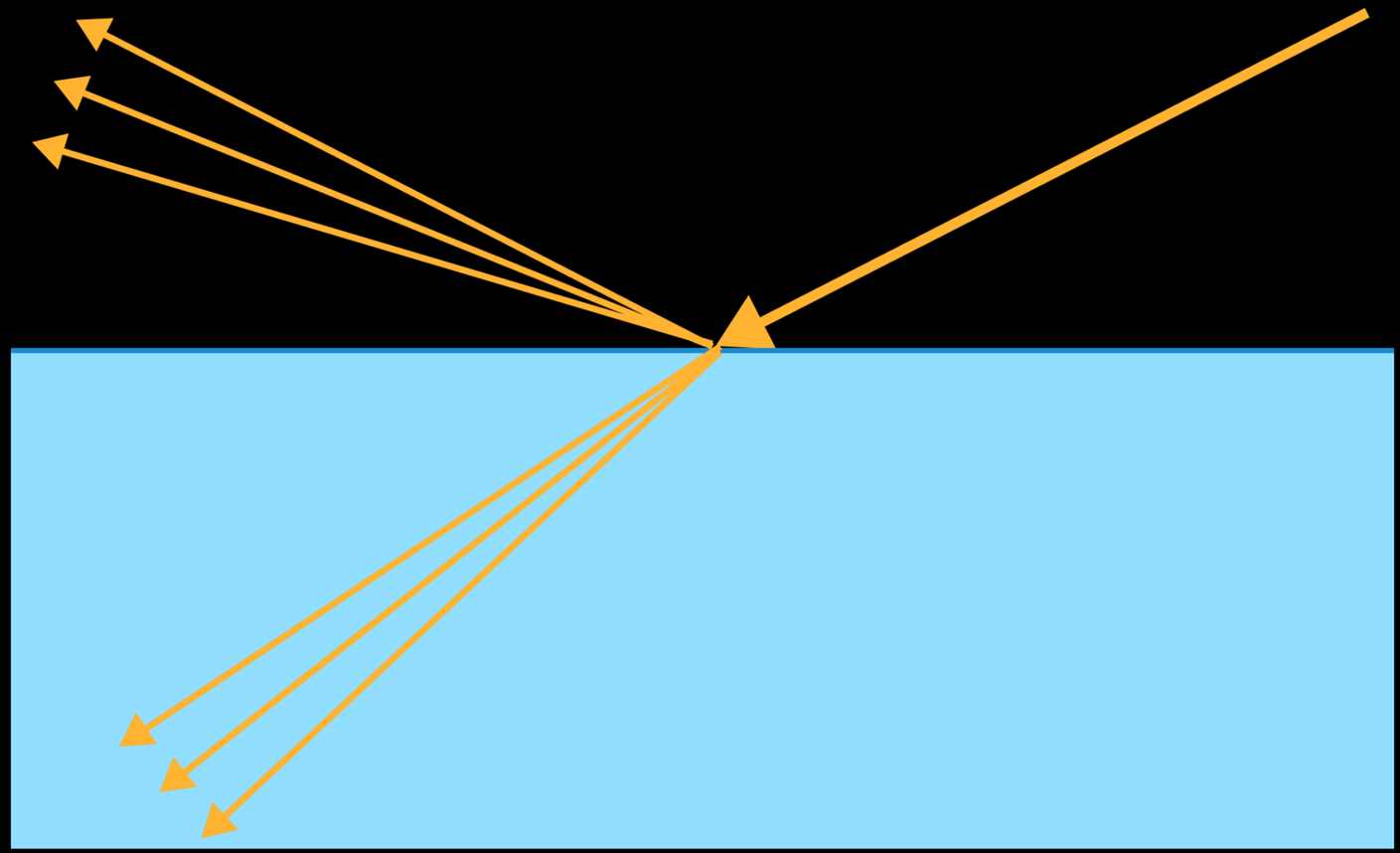
$$\int_{\Omega} f(x, w_i, w_o) \cos\theta \, dw_i \leq 1$$

Поверхность не может отразить больше света чем в нее пришло. Некоторая часть энергии может поглотиться

Расчет освещения

Взаимодействие света и поверхности

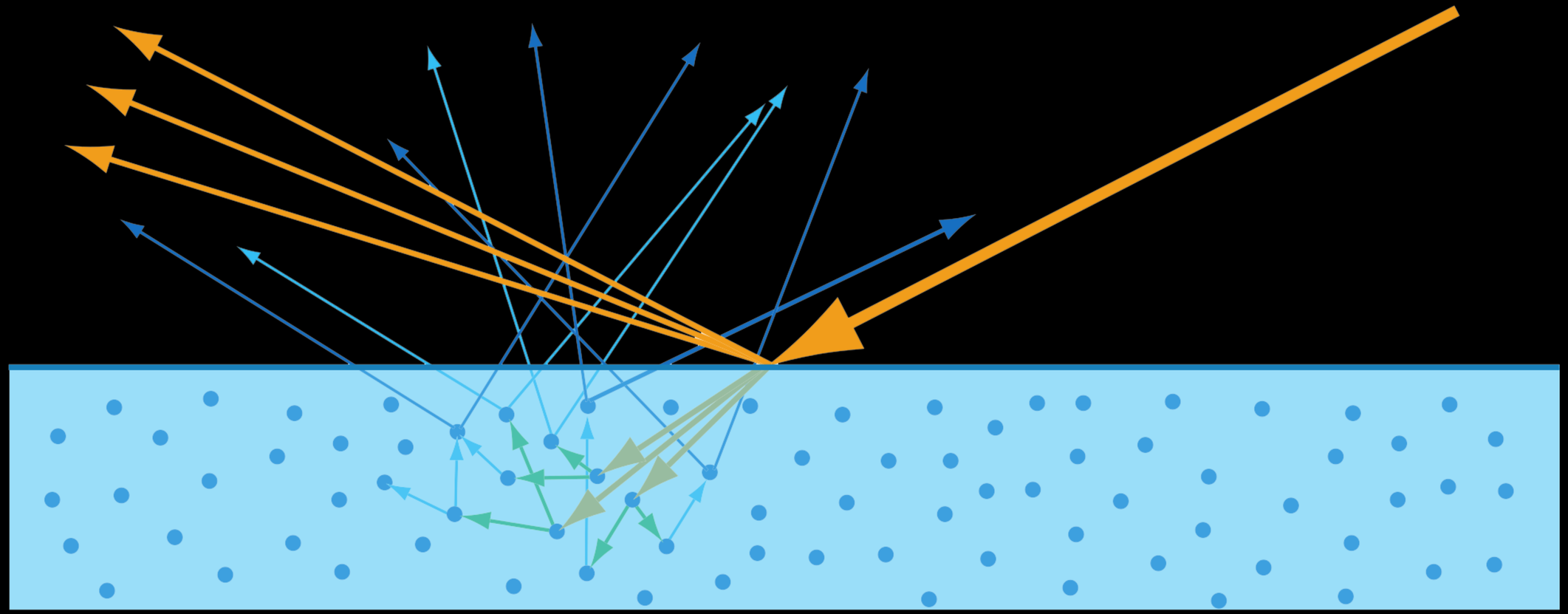
Несмотря на то что поверхность кажется плотной, свет проходит внутрь, преломляясь:



Расчет освещения

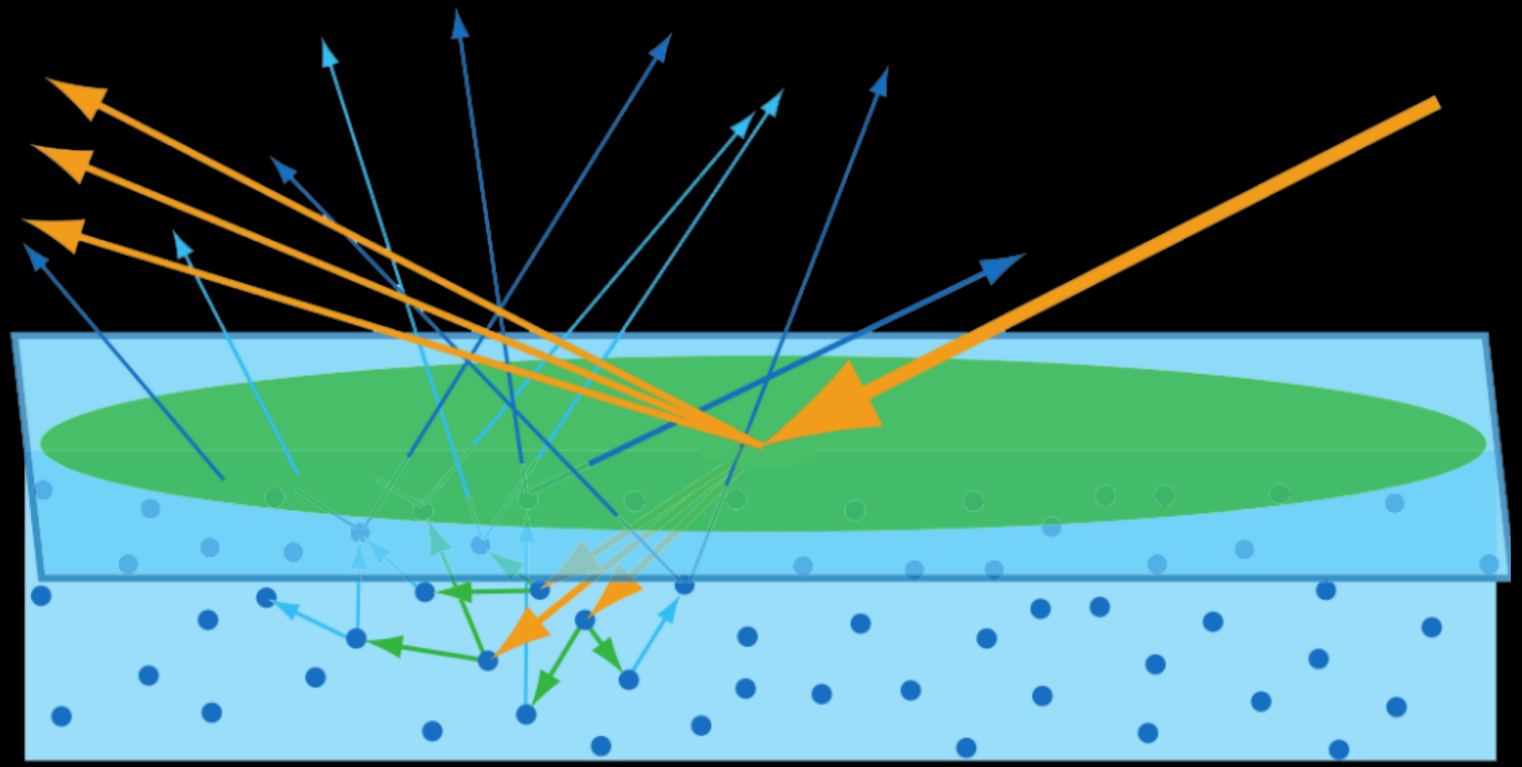
Взаимодействие света и поверхности

Свет многократно переотражается в верхних слоях поверхности, частично поглощаясь и выходит наружу:



Расчет освещения

Взаимодействие света и поверхности

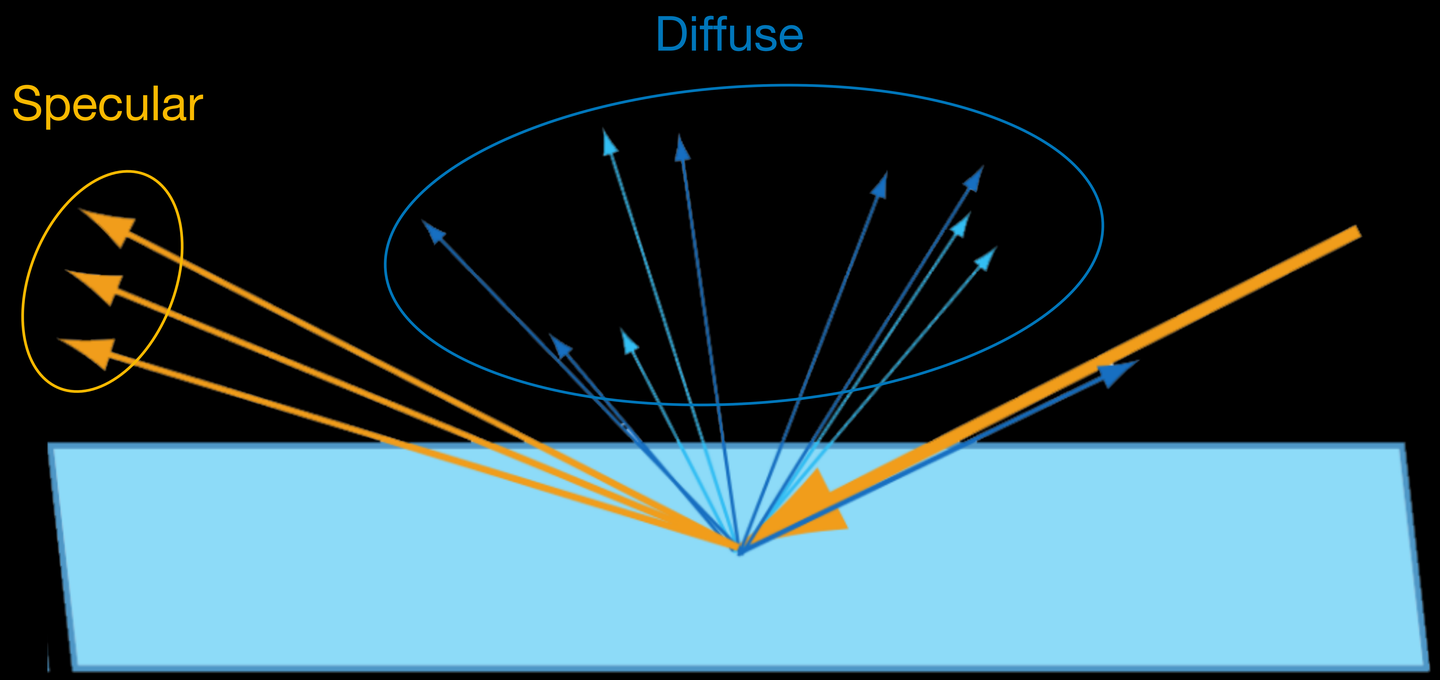


– Область шейдинга

Идея: можем пренебречь расстояниями выхода и считать всю область одной точкой

Расчет освещения

Виды отражений



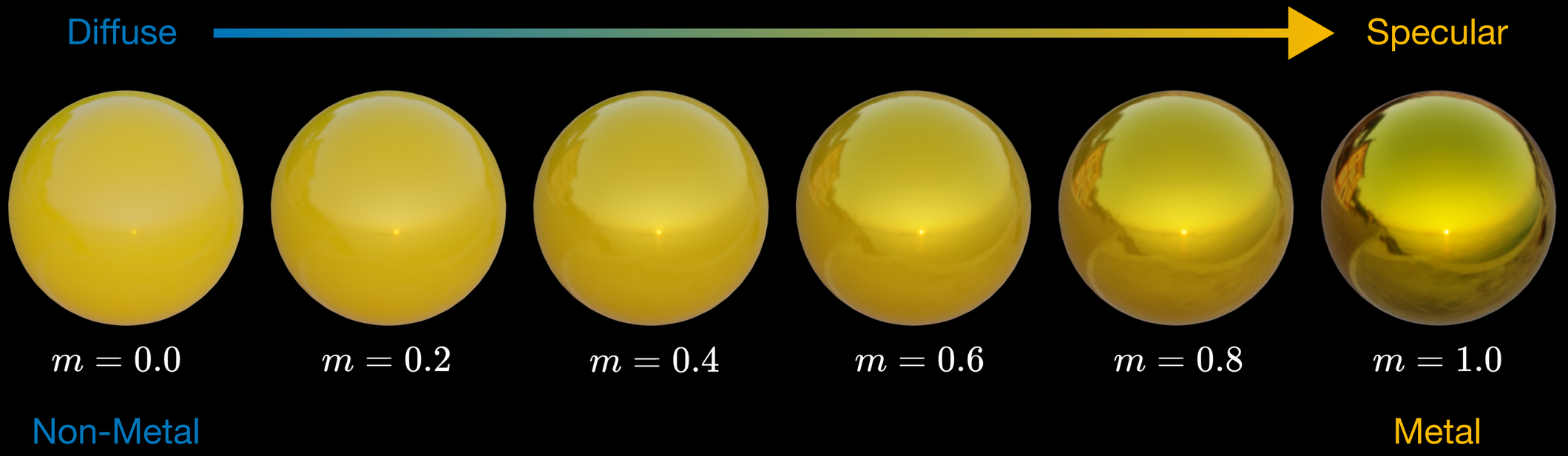
Specular – зеркальные отражения света

Diffuse – свет вышедший из поверхности вблизи от исходной точки

Расчет освещения

Виды отражений

Diffuse/Specular отражения проявляются в разной степени в зависимости от металлических свойств поверхности:



Расчет освещения

Specular reflection

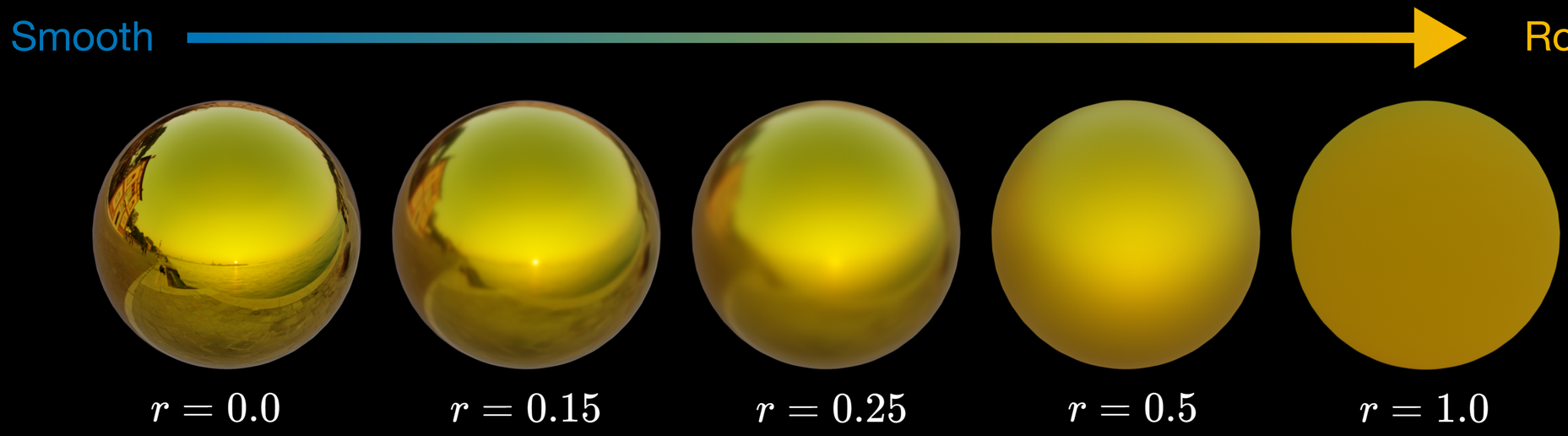




Расчет освещения

Specular reflection

Сфокусированность отражений зависит от гладкости поверхности на микроскопическом уровне (roughness):



Расчет освещения

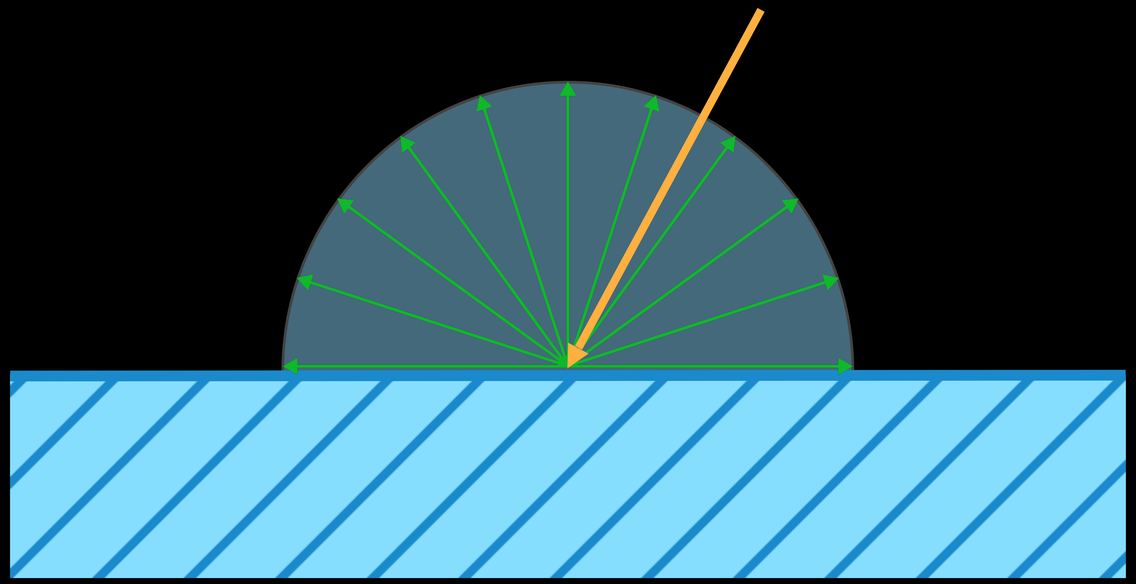
Lambertian diffuse model

$$f(x, w_i, w_o) = \frac{k_d}{\pi}$$

$$k_d \in [0; 1]^3$$

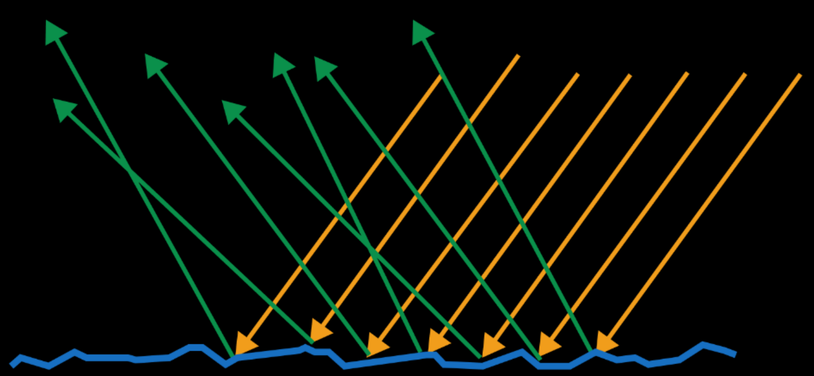
1. Яркость не зависит от того, под каким углом наблюдатель видит поверхность

2. Поверхность поглощает часть спектра света, т.е. имеет цвет (albedo или diffuse color)

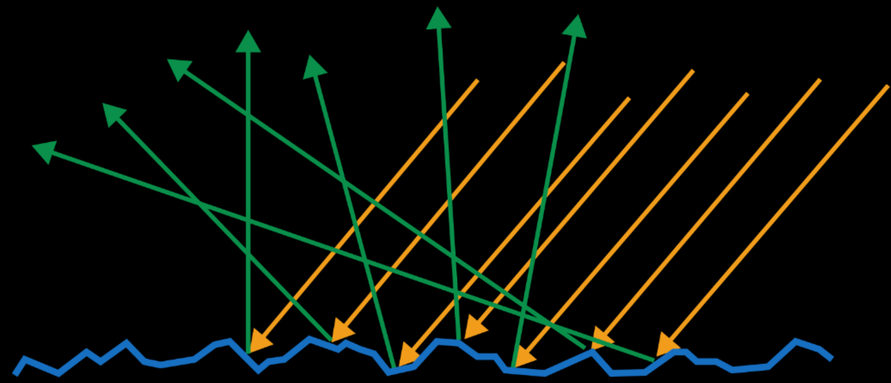


Расчет освещения

Specular reflection



Smoother surface



Rougher surface



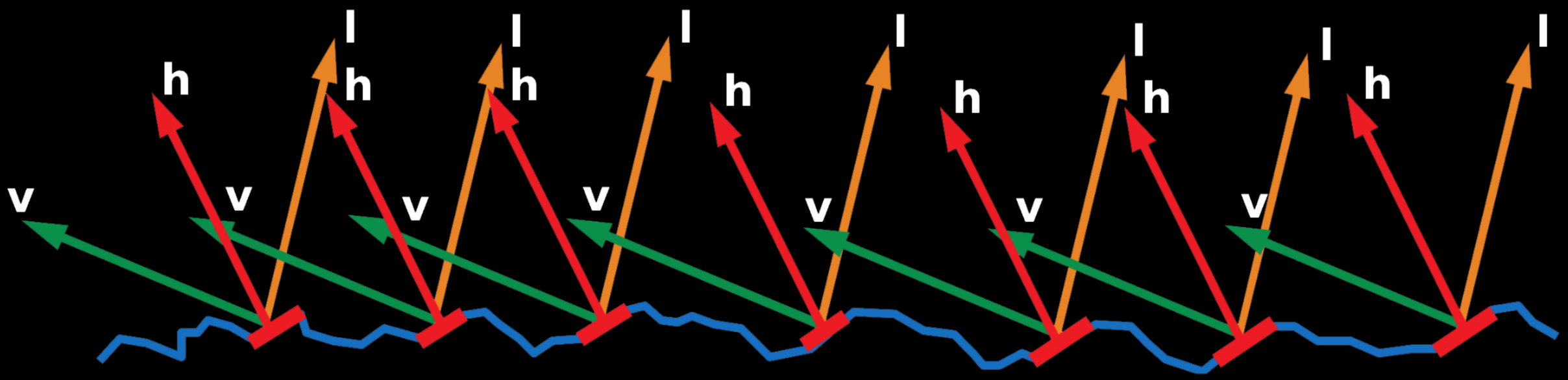


Расчет освещения

Microfacet theory

1. Specular отражения подчиняются закону “Угол падения = Углу отражения”
2. Неровности поверхности локально отклоняют нормаль поверхности h
3. Статистический подход к моделированию неровностей

$$h = \text{Halfway vector} = \frac{l + v}{2}$$



Расчет освещения

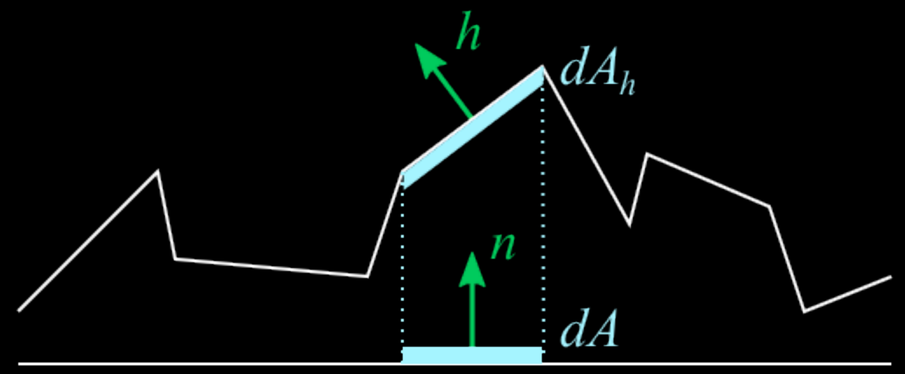
Normal **D**istribution **F**unction

Показывает насколько вероятна определенная ориентация microfacet нормали

Накладывается требование неотрицательности и нормированности:

$$D(h) \geq 0$$

$$\int_{\Omega} D(h) \langle h, n \rangle \, dw_h = 1$$



Скалярное произведение в условии возникает из-за проекции дифференциала площади

Расчет освещения

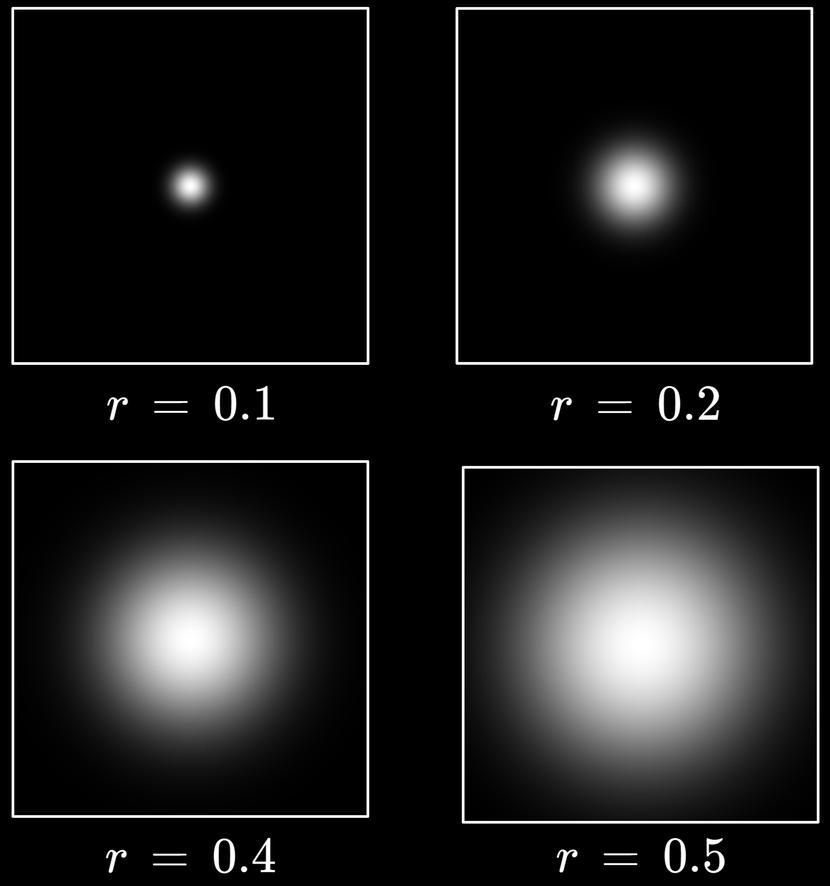
Beckmann NDF

Определяется гауссовым распределением
с константой нормализации:

$$D = \frac{1}{\pi\alpha^2 \cos^4 \theta_h} e^{-\frac{\tan^2 \theta_h}{\alpha^2}}$$

θ_h – угол между h и n

$\alpha = r^2$ – шероховатость поверхности

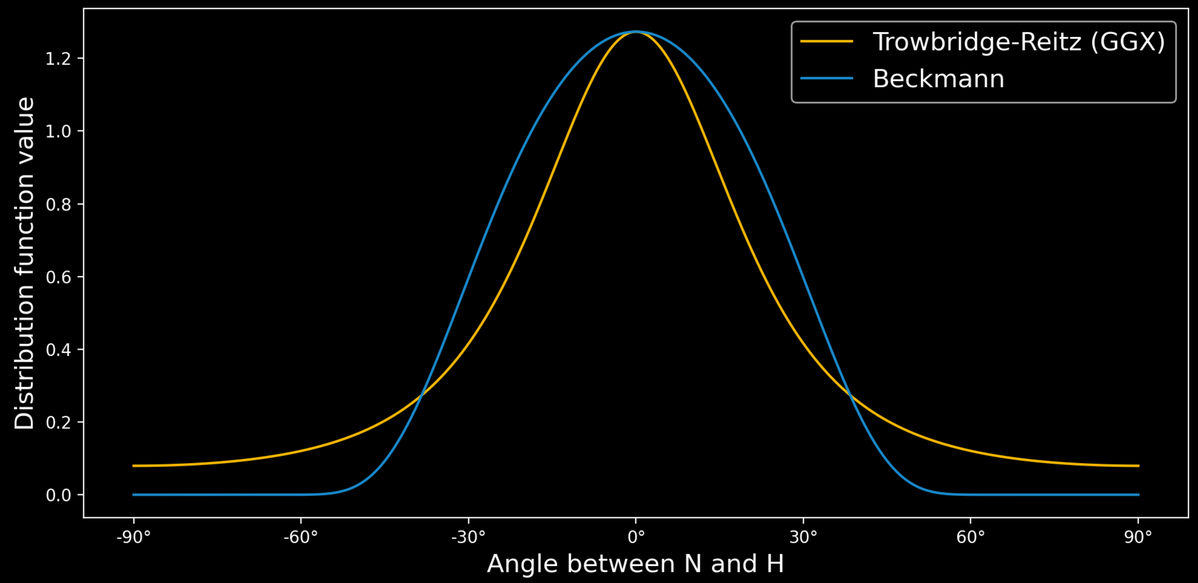


Расчет освещения

Trowbridge-Reitz (GGX) NDF

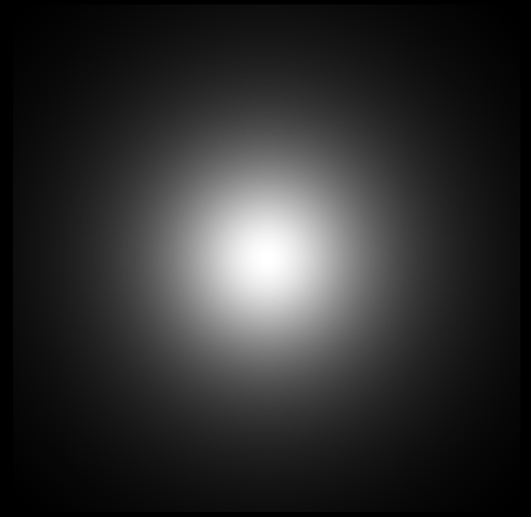
Распределение имеет тяжелые хвосты и более сфокусированный засвет:

$$D = \frac{\alpha}{\pi((\alpha - 1)\cos\theta_h + 1)^2}$$





Beckmann NDF

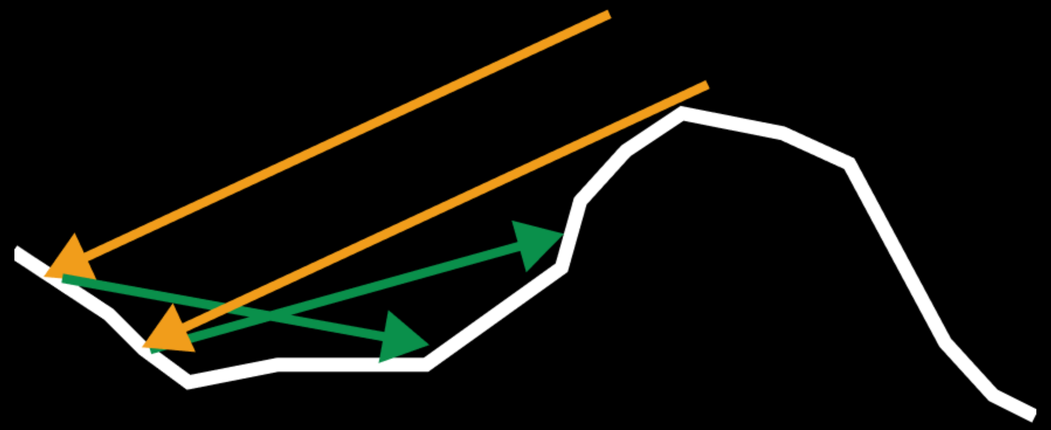


GGX

Расчет освещения

Geometry function

Свет, падающий на поверхность,
может отразиться в microfacet

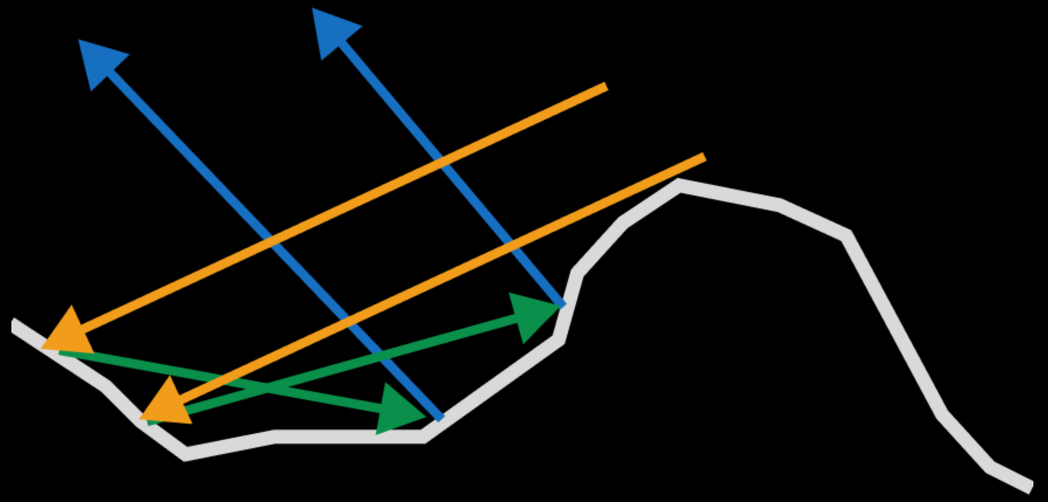


Расчет освещения

Geometry function

Свет, падающий на поверхность, может отразиться в microfacet

Некоторая часть света должна покинуть поверхность



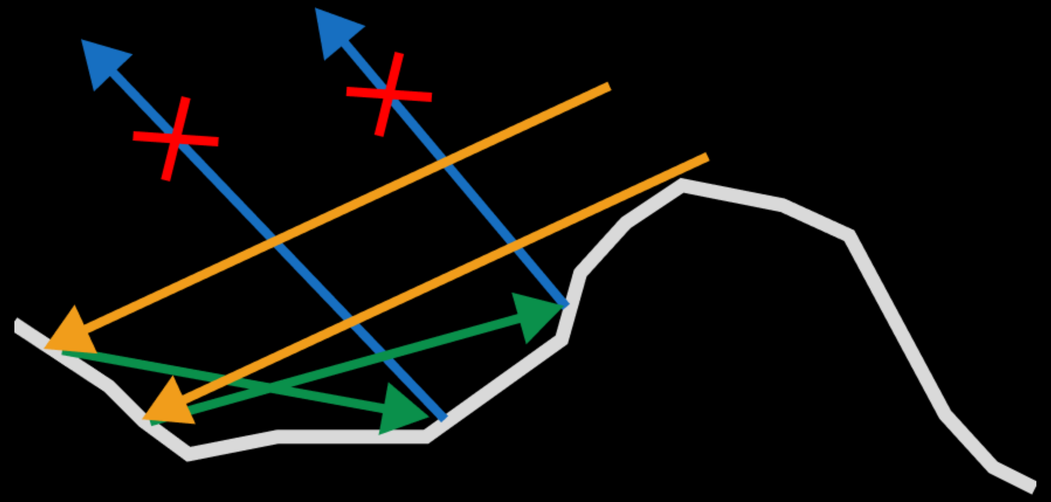
Расчет освещения

Geometry function

Свет, падающий на поверхность, может отразиться в microfacet

Некоторая часть света должна покинуть поверхность

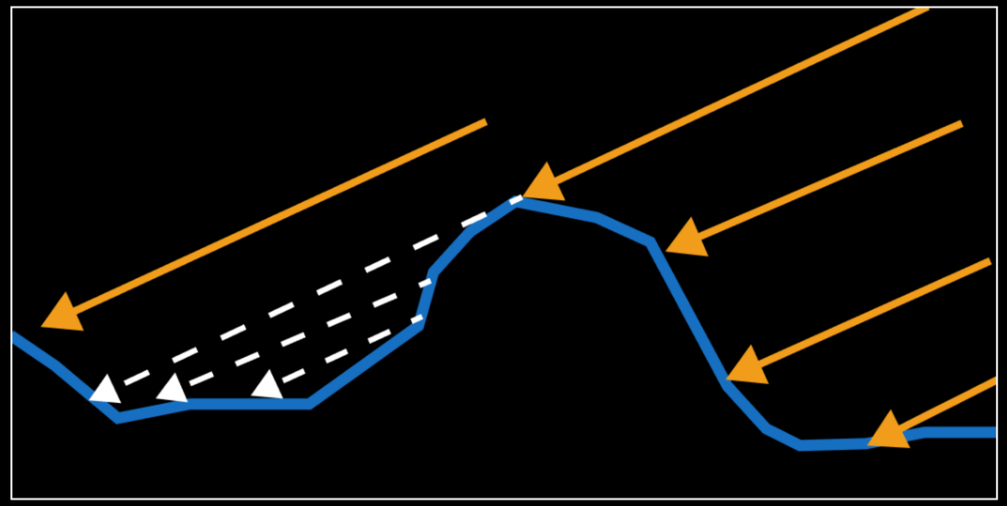
Однако, microfacet theory подразумевает, что такой свет будет поглощен полностью



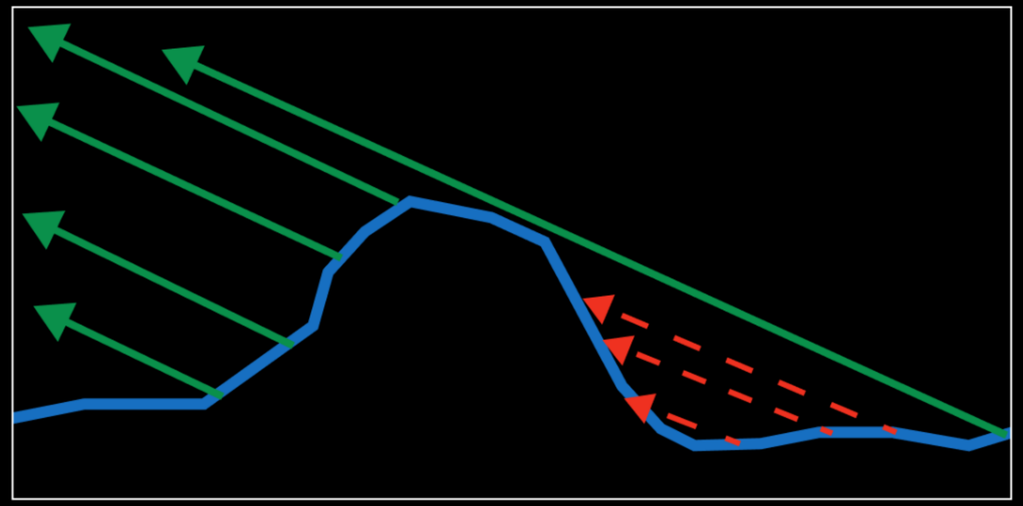
Расчет освещения

Geometry function

Эффект симметричен для падающего и отраженного света:



Shadowing



Masking

Расчет освещения

Geometry function

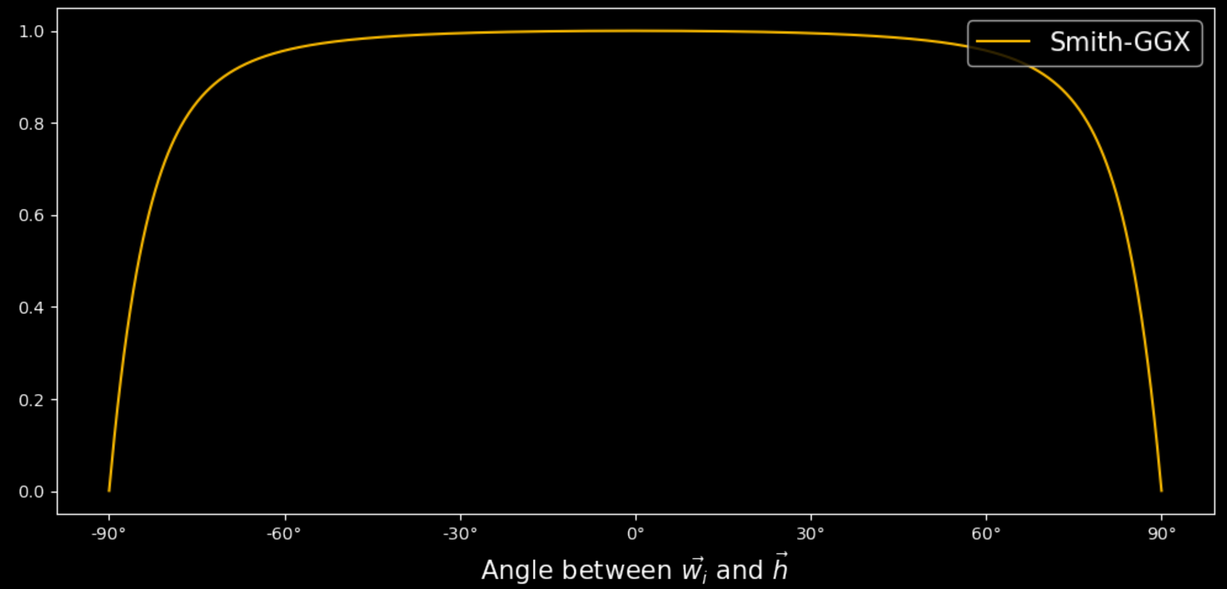
Должна удовлетворять свойствам:

1. Нормированность:

$$0 \leq G(w_i, w_o, h) \leq 1$$

1. Симметричность (обратимость Гельмгольца):

$$G(w_i, w_o, h) = G(w_o, w_i, h)$$



Correlated Smith-GGX:

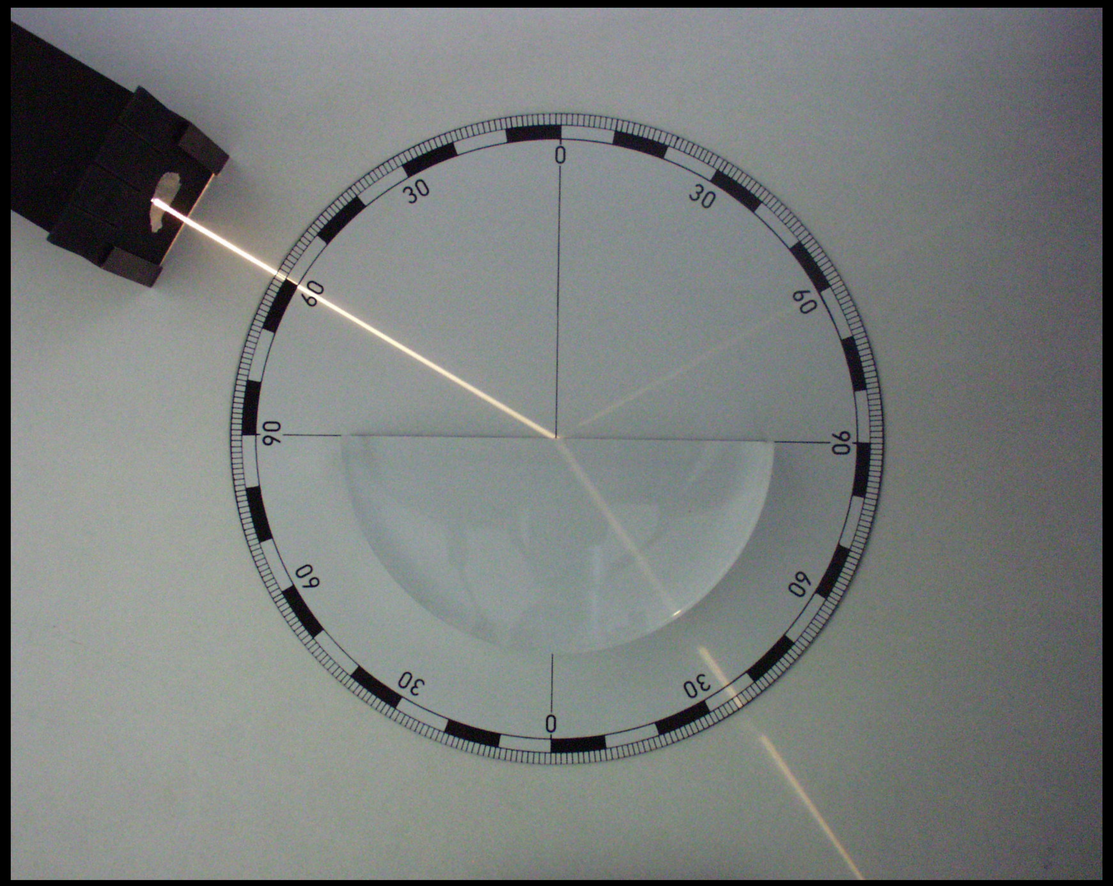
$$G = \frac{1}{1 + G_1(w_i, h) + G_1(w_o, h)}$$

$$G_1 = \frac{1}{2\left(\sqrt{1 + \alpha^2 \tan^2 \theta} - 1\right)}$$

Расчет освещения

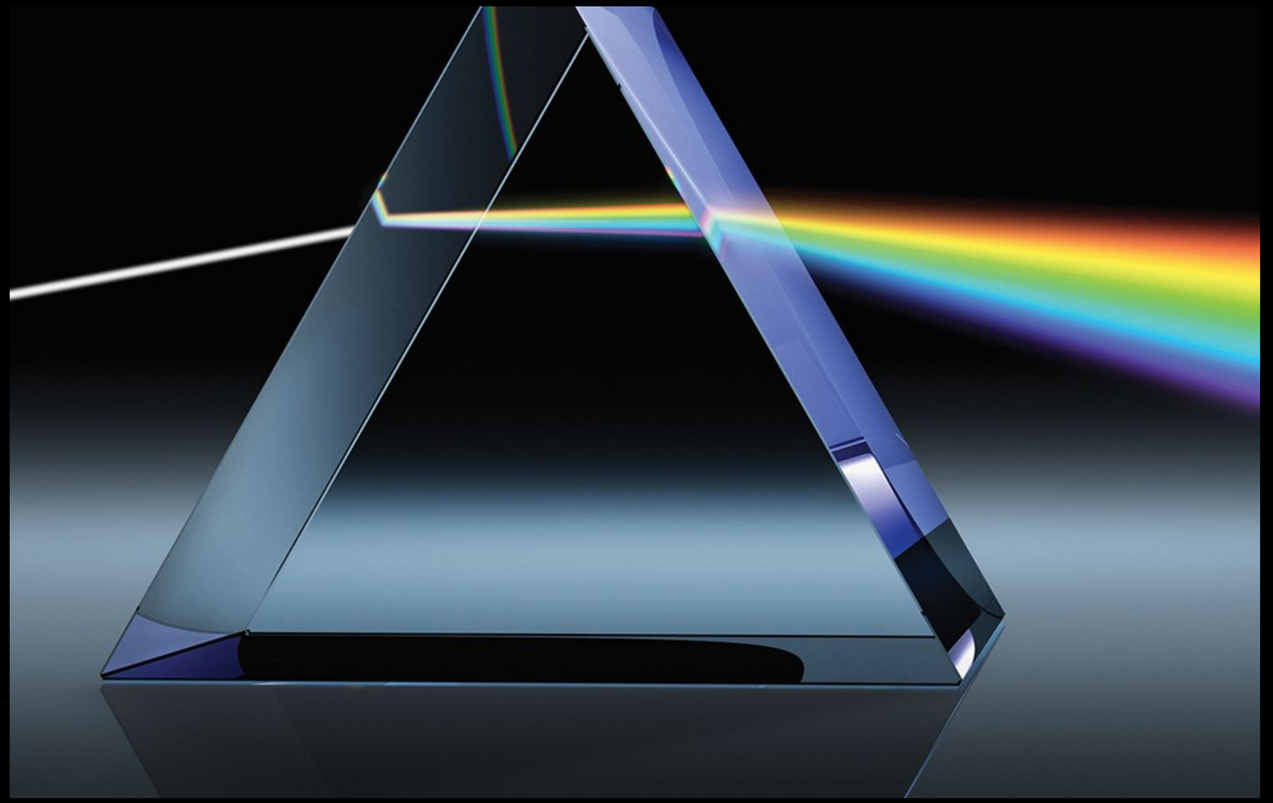
Fresnel reflectance

Свет, падающий на поверхность, одновременно отражается и преломляется в разных пропорциях



Расчет освещения

Fresnel reflectance



Расчет освещения

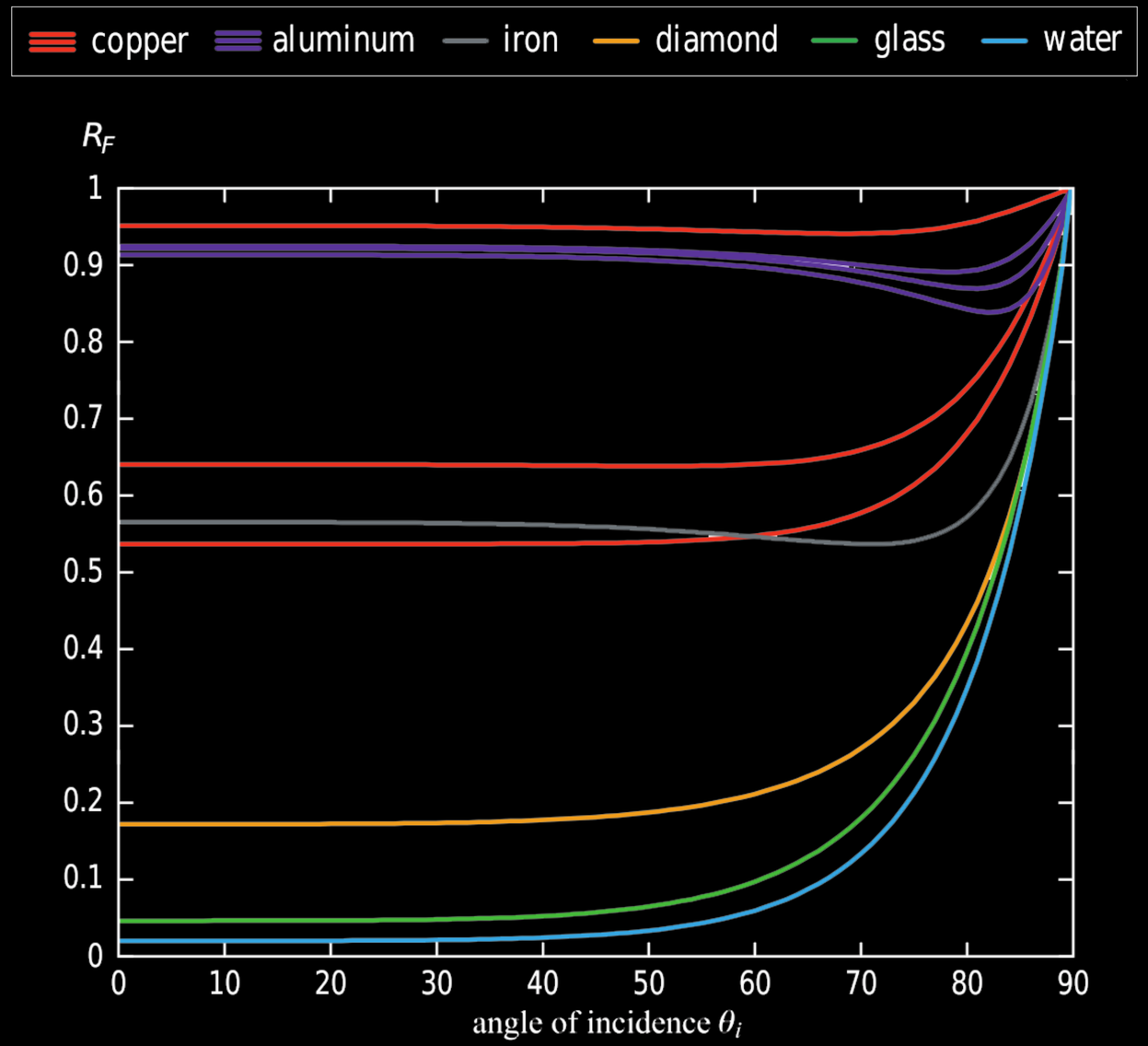
Fresnel reflectance

Уравнение Френеля описывает вероятность отражения света:

$$F = \left| \frac{n_1 \cos\theta - n_2 \sqrt{1 - \left(\frac{n_1}{n_2}\sin\theta\right)^2}}{n_1 \cos\theta + n_2 \sqrt{1 - \left(\frac{n_1}{n_2}\sin\theta\right)^2}} \right|^2$$

n_1, n_2 – коэффициенты преломления

В общем случае зависят от длины волны!



Расчет освещения

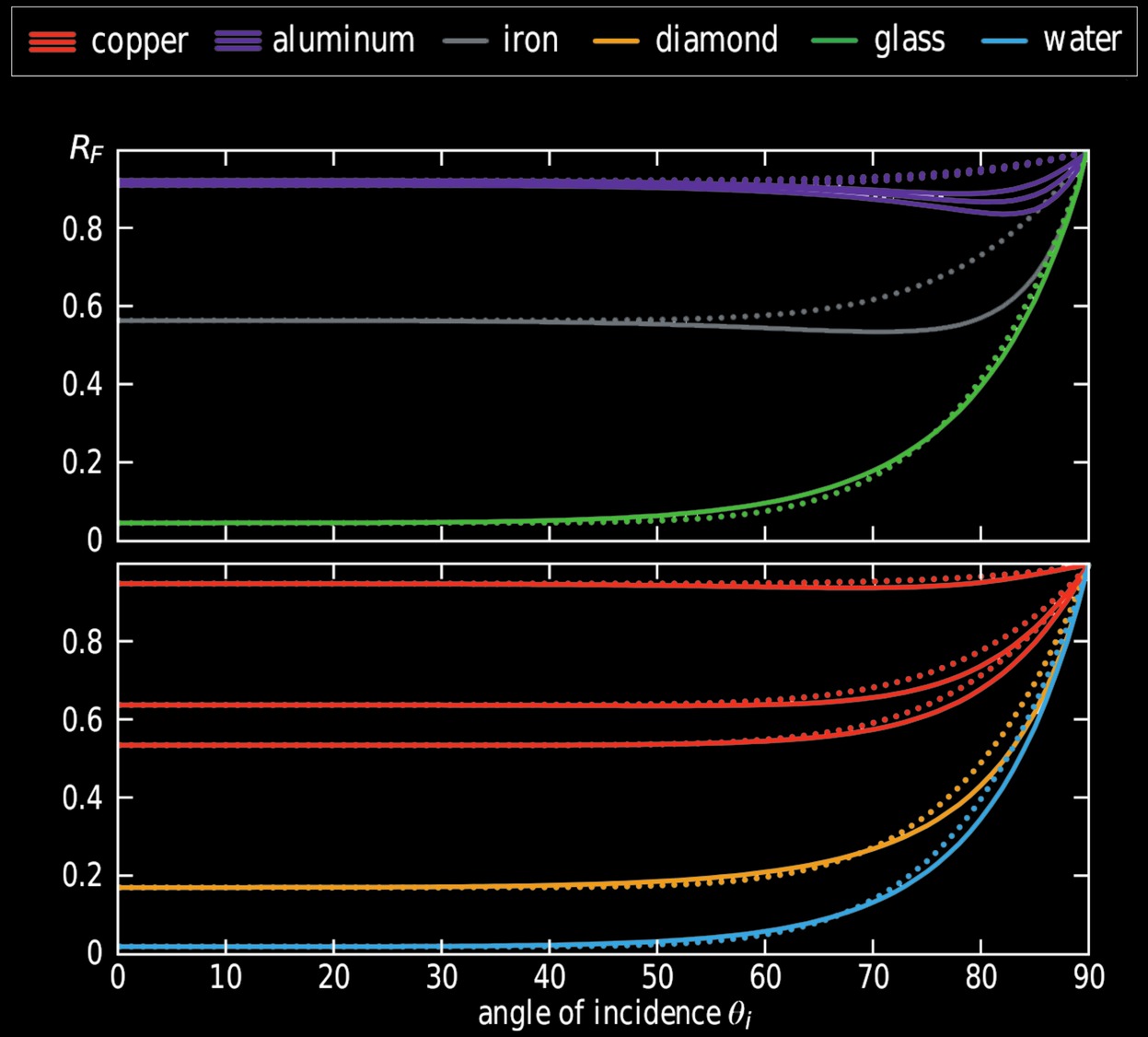
Fresnel-Schlick formula

На практике используется аппроксимация:

$$F(l,\, h) \;=\; F_0 \;+\; (1 - F_0)(1 - \cos\theta)^5$$

F_0 – цвет отражения в направлении нормали

θ – угол падения света



Расчет освещения

Reflectance color

Цвет отражения диэлектриков монохроматический, обычно принимается равным 0.04

На практике рассчитывается как:

$$F_0 = 0.04\,(1 - m) + k_d\,m$$

$m \in [0; 1]$ – металлические св-ва

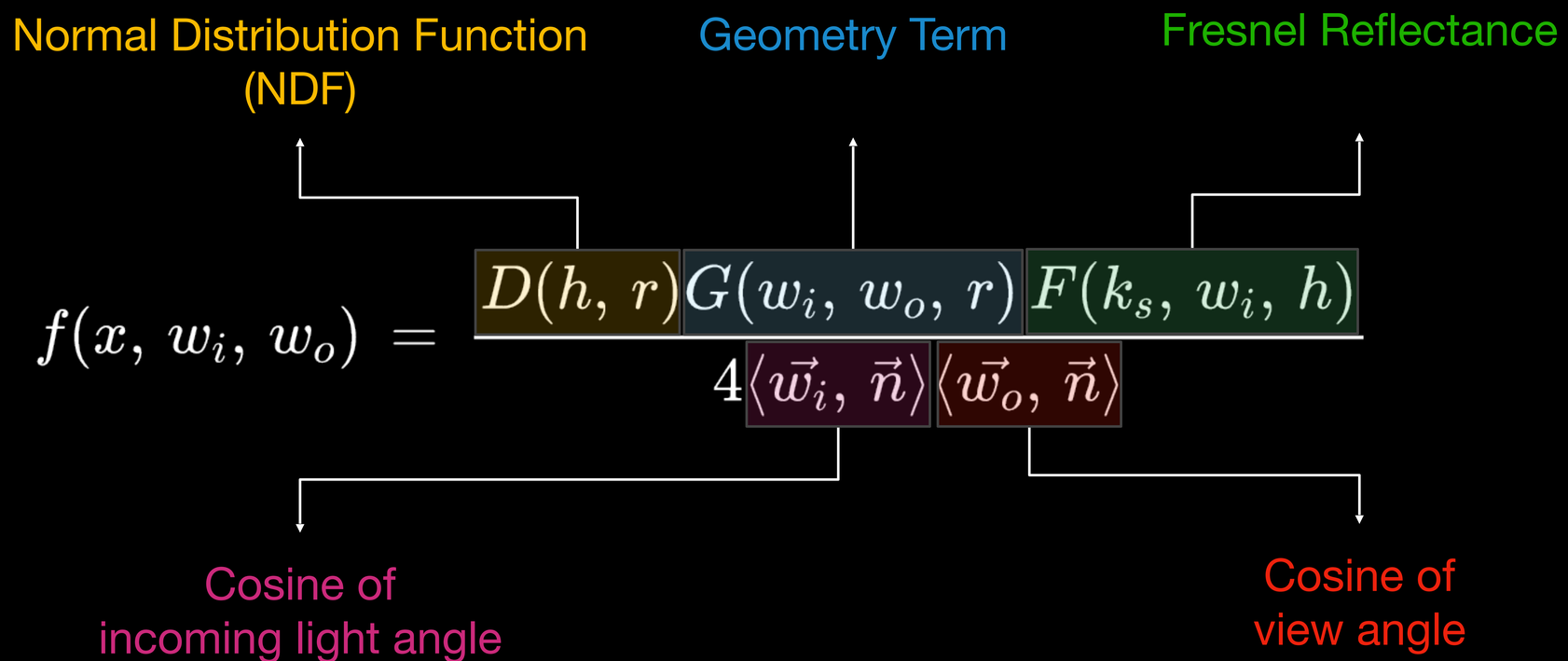
k_d – базовый цвет

Dielectric	F_0 (Linear, Float)	F_0 (sRGB, U8)	Color
Water	0.020	39	
Plastic, Glass	0.040 – 0.045	56 – 60	
Crystalware, Gems	0.050 – 0.080	63 – 80	
Diamond-like	0.100 – 0.200	90 – 124	

Metal	$R_F(0°)$ **(Linear)**	$R_F(0°)$ **(sRGB)**	**Color**
Gold	1.00,0.71,0.29	1.00,0.86,0.57	
Silver	0.95,0.93,0.88	0.98,0.97,0.95	
Copper	0.95,0.64,0.54	0.98,0.82,0.76	
Iron	0.56,0.57,0.58	0.77,0.78,0.78	
Aluminum	0.91,0.92,0.92	0.96,0.96,0.97	

Расчет освещения

Cook-Torrance BRDF



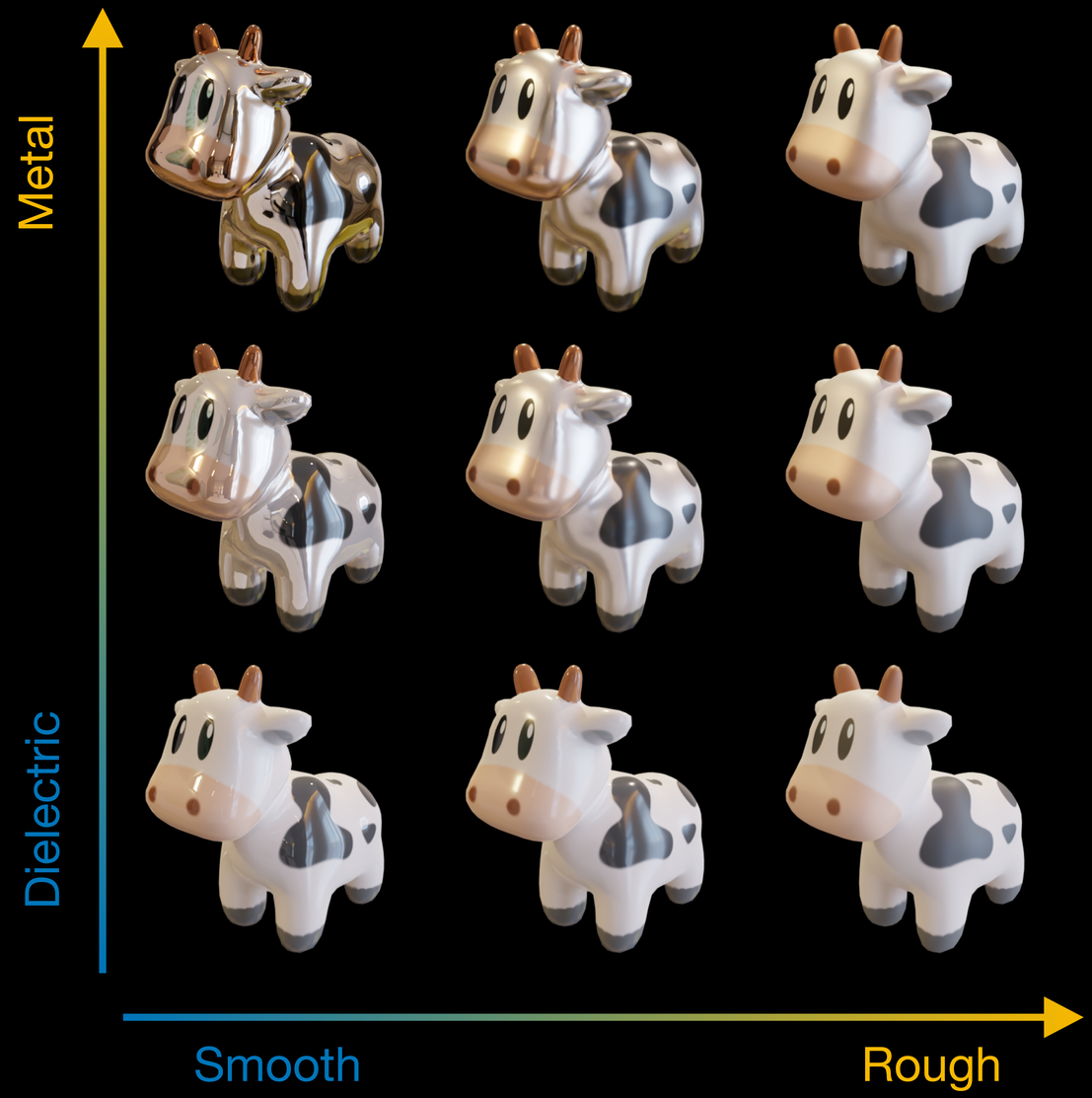
Расчет освещения

Simplified PBR

PBR = Physically Based Rendering

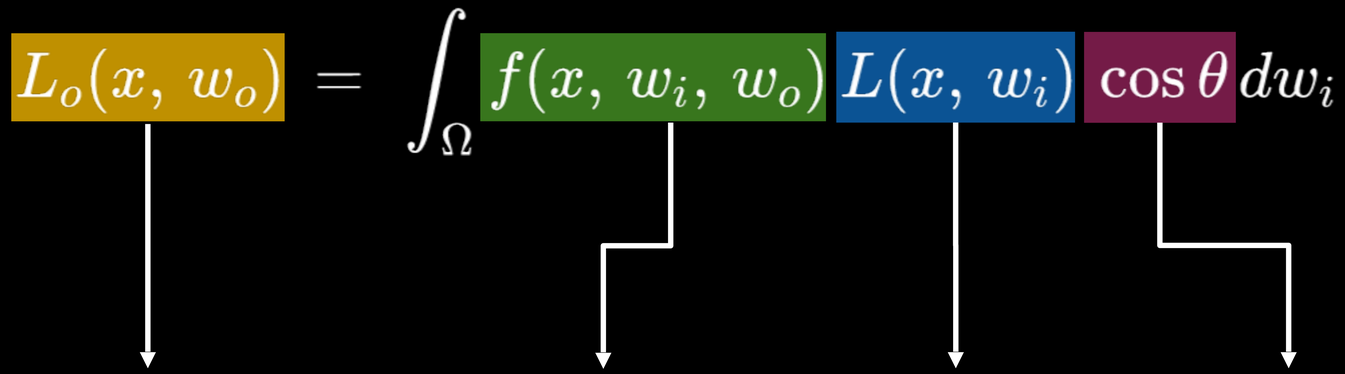
Материал описывается с помощью:

- Base color
- Roughness
- Metallic



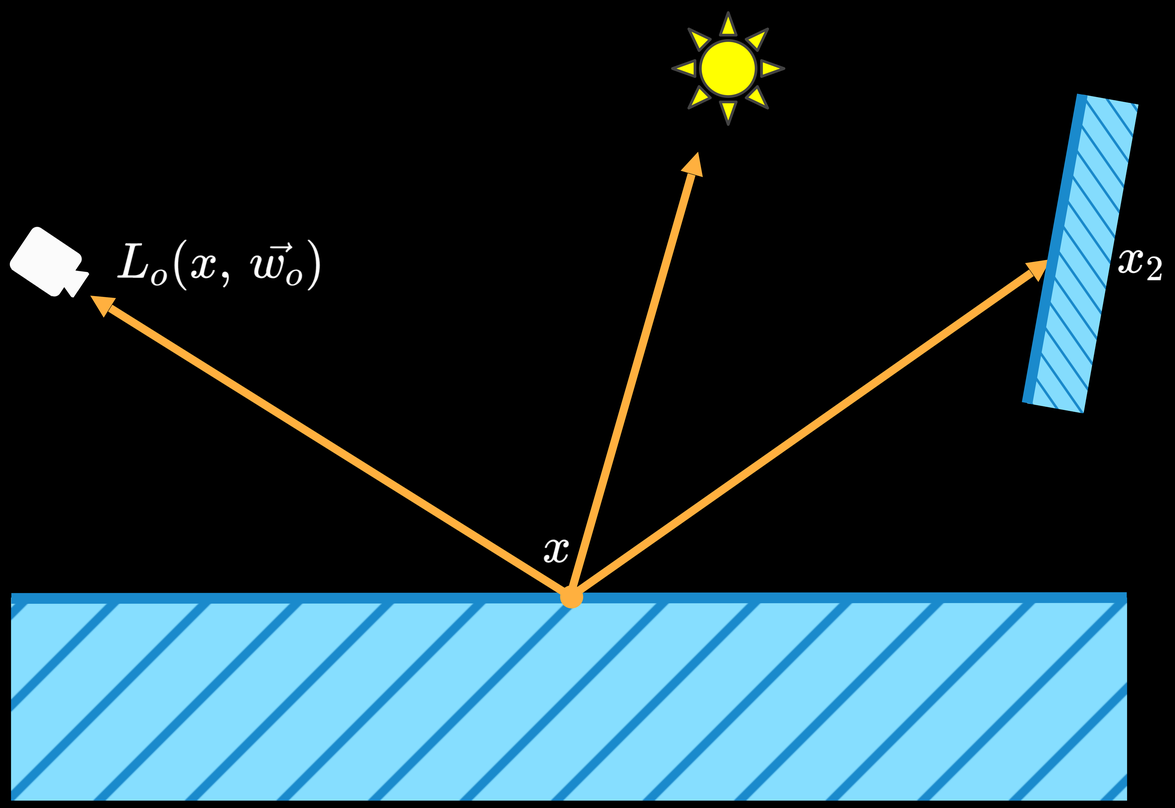
Расчет освещения

Rendering equation



Расчет освещения

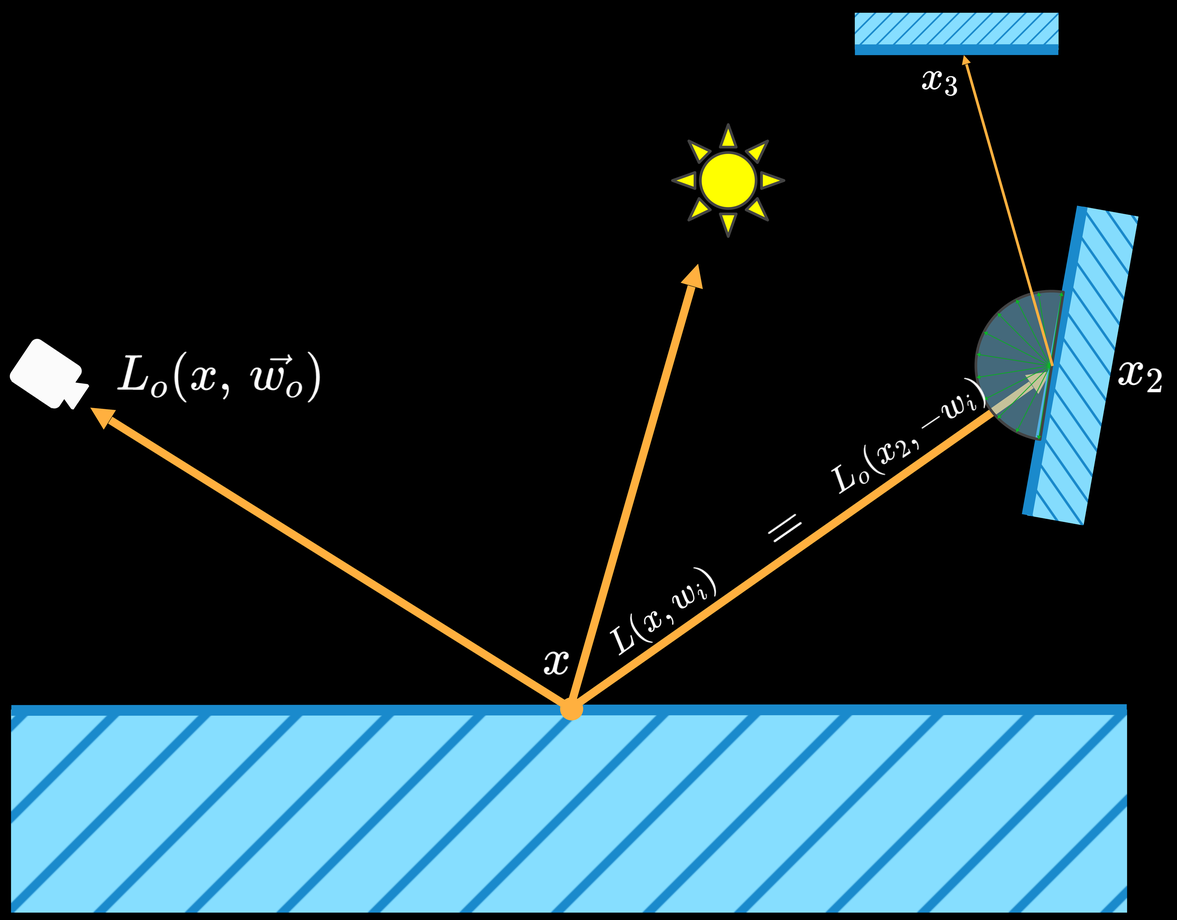
Rendering equation



Расчет интеграла рендеринга включает как источники света напрямую (direct light), так и свет отраженный от других объектов в сцене (indirect light)

Расчет освещения

Rendering equation



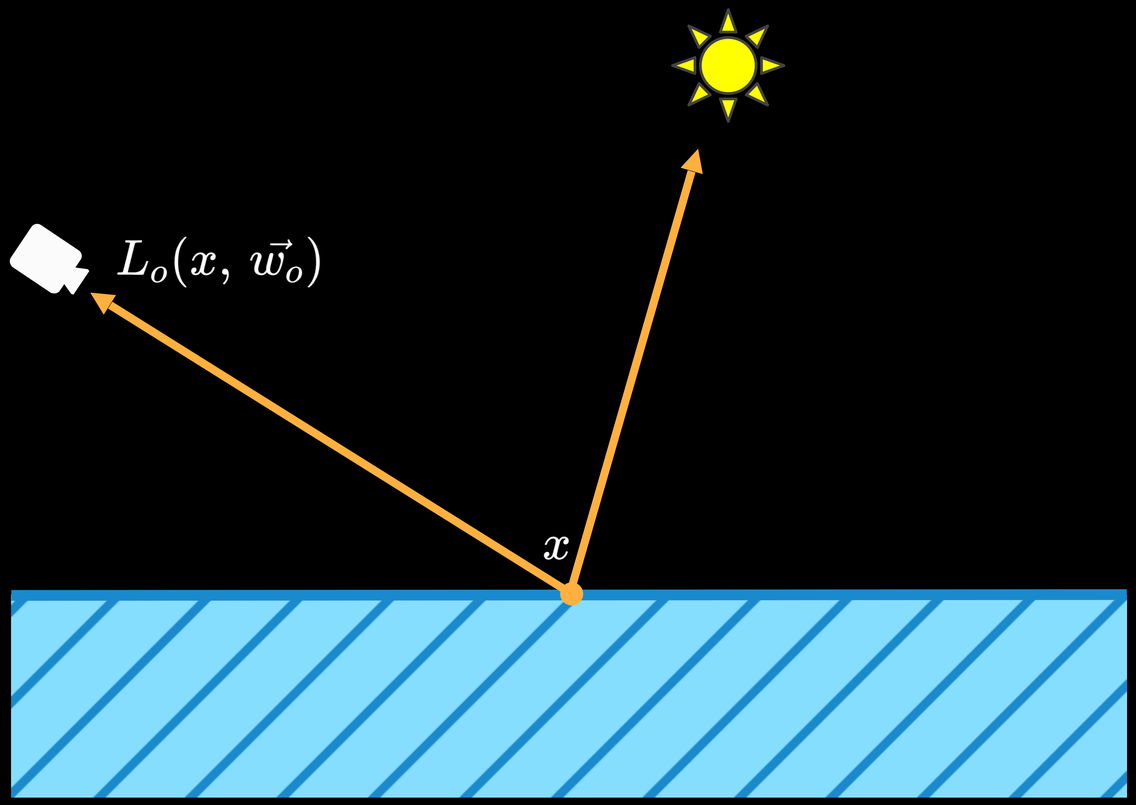
Расчет интеграла рендеринга включает как источники света напрямую (direct light), так и свет отраженный от других объектов в сцене (indirect light)

Rendering equation – бесконечномерный интеграл

Как рассчитать?

Расчет освещения

Rendering equation



Способ 1. Источник света точечный, непрямого света не существует

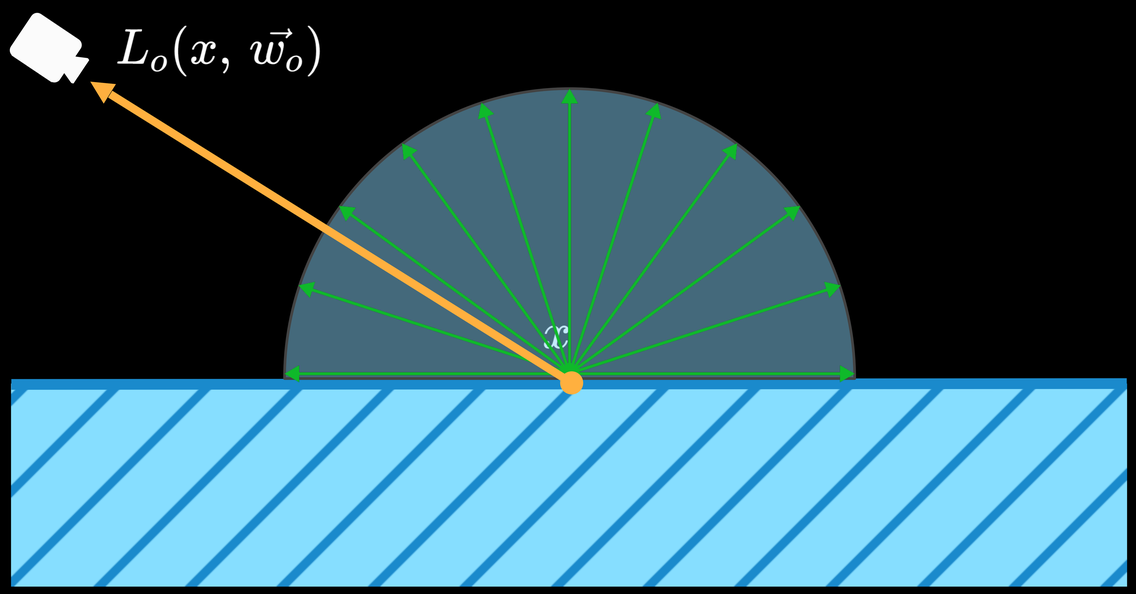
$$L_o(x, w_o) \approx f(x, w_i, w_o)\, L(w_i)\, \cos\theta$$

Пример: nvdiffmodeling



Расчет освещения

Rendering equation



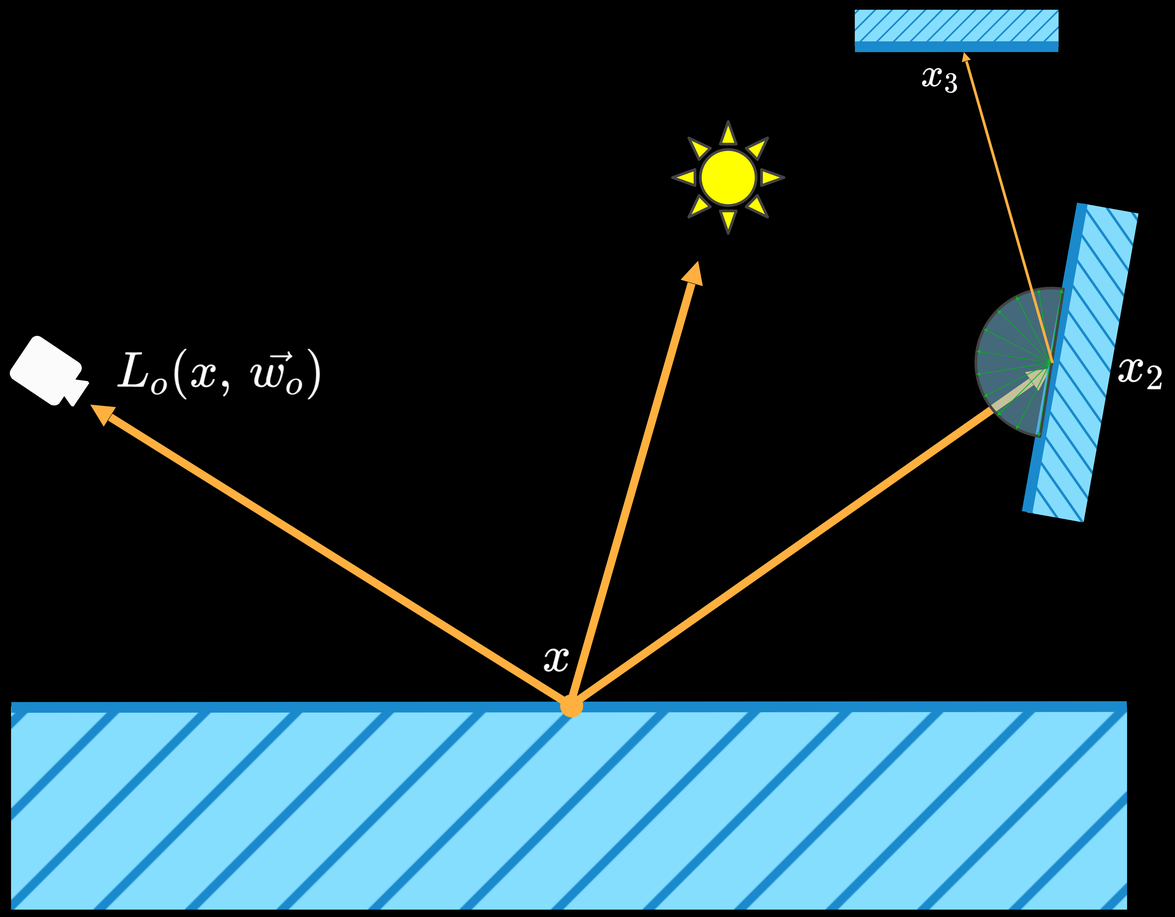
Способ 2. Свет приходит со всех сторон, непрямого освещения нет. Split-sum аппроксимация интеграла

Пример: nvdiffrec



Расчет освещения

Rendering equation



Способ 3. Monte Carlo оценка интеграла, полная симуляция переноса света

Примеры: Redner, Mitsuba



Расчет освещения

Summary

1. Расчет освещения происходит путем оценки интеграла рендеринга

2. Мы рассмотрели два вида взаимодействия света с поверхностью – diffuse и specular reflections

2. Взаимодействие света с поверхностью зависит от физических свойств поверхности (например, diffuse color, roughness, metallic). Математически описывается при помощи BRDF функций

3. Точный расчет интеграла рендеринга невозможен. Используются аппроксимации различной сложности

Neural Volumetric Rendering



Neural Volumetric Rendering

Расчет света вдоль луча, проходящего через 3D пространство



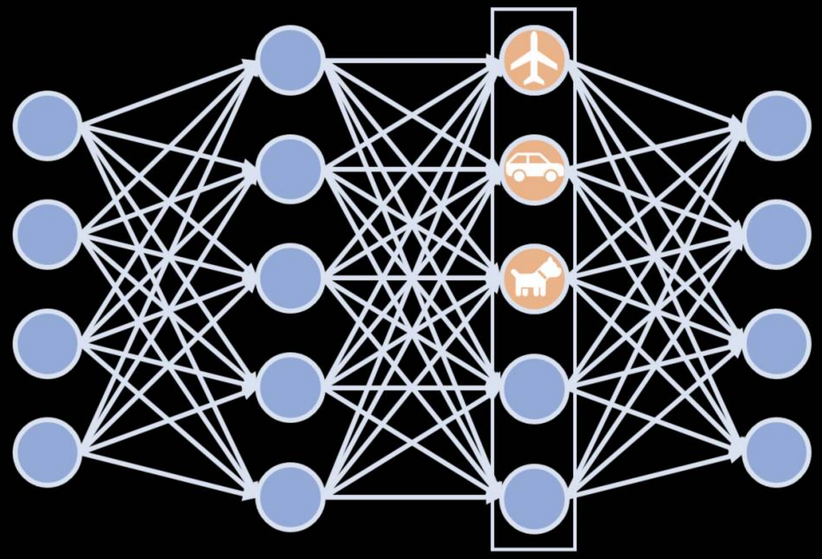
Neural Volumetric Rendering

Модель рендеринга, основанная на представлении сцены в виде объема заполненного бесконечно малыми частицами, и не включающая расчет пересечения луча с поверхностью



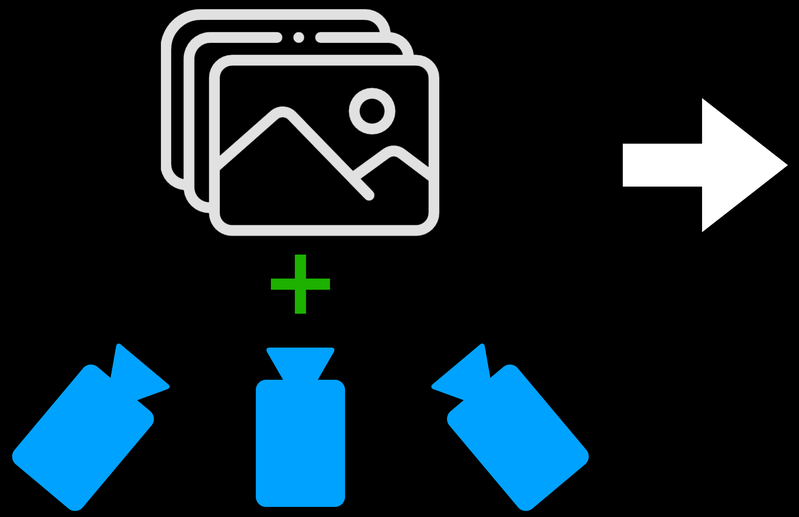
Neural Volumetric Rendering

Все, что есть в сцене, представляется в виде одной MLP — непрерывного и дифференцируемого представления



Мотивация

Novel view synthesis





Задача: Синтезировать изображение с нового произвольного ракурса используя фиксированный набор изображений с позами камер

Roadmap

- Volumetric formulation for Neural Radiance Filed (NeRF)
- Implicit neural scene representation
- NeRF summarised
- Application & Extensions

$$L_o(x, w) = ??$$

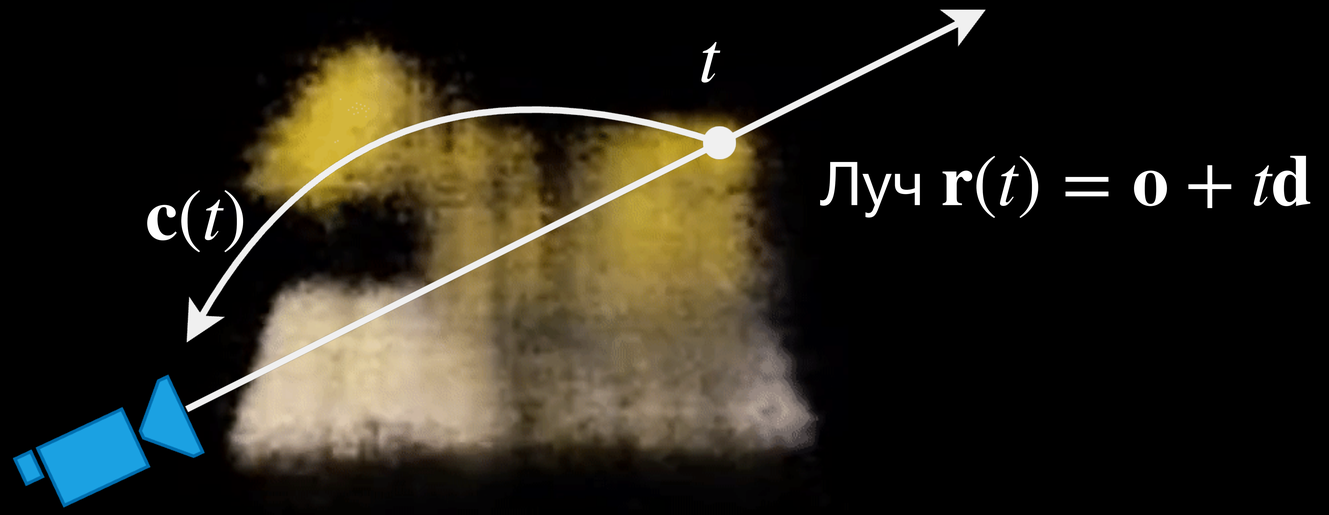
Volumetric Rendering

Volumetric formulation for NeRF



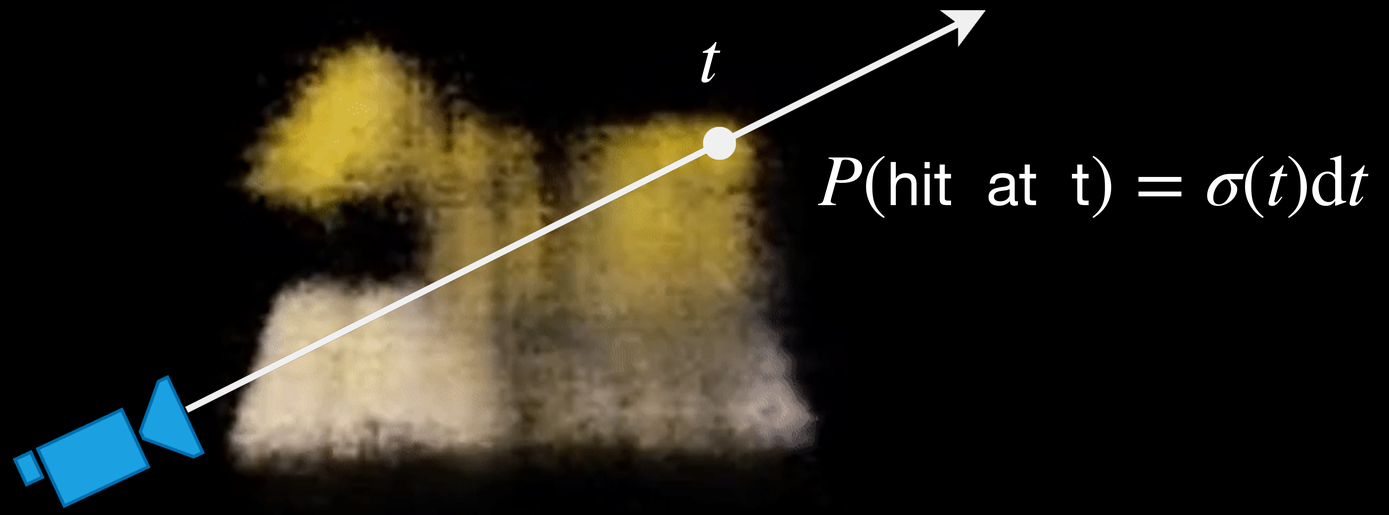
Сцена представлена объемом,
заполненным цветными частицами

Volumetric formulation for NeRF



Если луч $\mathbf{r}(\mathbf{t})$ не поглощается частицей
в положении t, то мы аккумулируем ее
цвет $\mathbf{c}(t)$ в пиксель

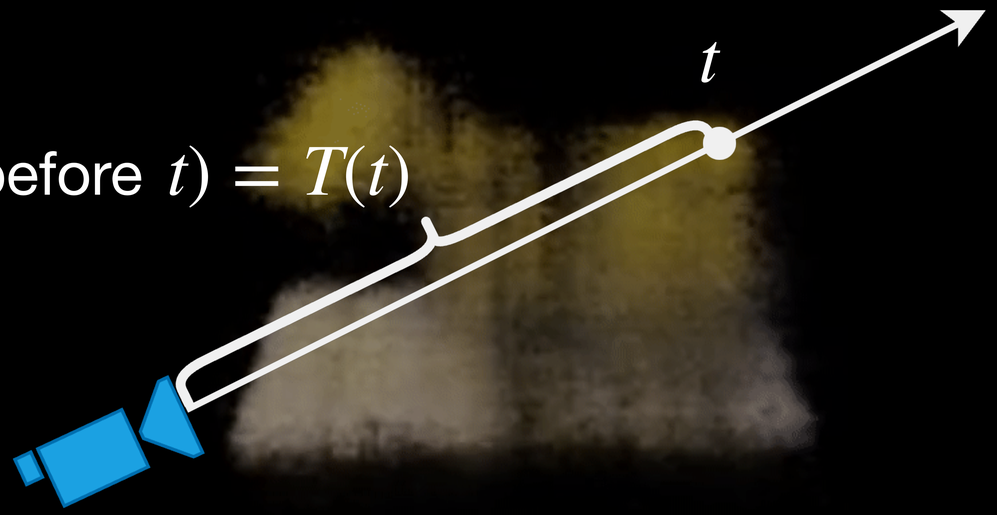
Volumetric formulation for NeRF



Модель поглощения вероятностная: вероятность поглощения энергии в окрестности точки t равна $\sigma(t)\mathrm{d}t$

$\sigma(t)$ —volume density

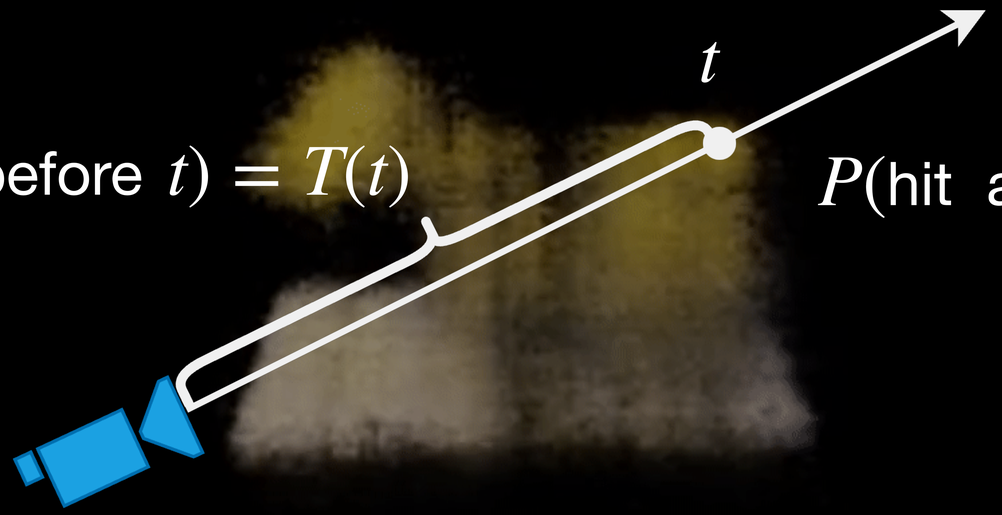
Volumetric formulation for NeRF



Вероятность того, что на отркезке $[t_1, t]$ не произошло поглощения энергии задается величиной $T(t)$

$T(t)$ — transmittance

Volumetric formulation for NeRF



$\sigma(t)$ и $T(t)$ связаны законом вероятности:

$$P(\text{no hits before } t + \mathrm{d}t) = P(\text{no hits before } t) \times P(\text{no hits at } t)$$

$$T(t + \mathrm{d}t) = T(t)(1 - \sigma(t))\mathrm{d}t$$

Volumetric formulation for NeRF

$$T(t + \mathrm{d}t) = T(t)(1 - \sigma(t))\mathrm{d}t$$

$$T(t) + T'(t)\mathrm{d}t = T(t) - T(t)\sigma(t)\mathrm{d}t \quad \text{<= Применим Тейлора слева}$$

$$\frac{T'(t)}{T(t)}\mathrm{d}t = -\sigma(t)\mathrm{d}t \quad \text{<= Перегруппируем}$$

$$\log T(t) = -\int_{t_0}^{t} \sigma(s)\mathrm{d}s \quad \text{<= Проинтегрируем} \quad T(t) = \exp\left(-\int_{t_0}^{t} \sigma(s)\mathrm{d}s\right)$$

Volumetric formulation for NeRF

Вероятность того, что свет поглотится в точке t

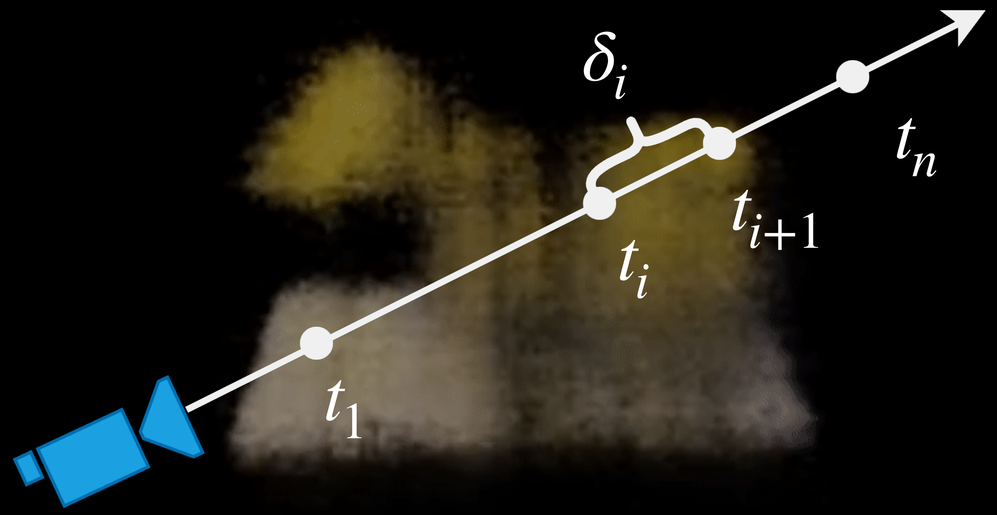
$$T(t)\sigma(t)\mathrm{d}t = \exp\left(-\int_{t_0}^{t}\sigma(s)\mathrm{d}s\right)\sigma(t)\mathrm{d}t$$

Мат. ожидание цвета в данном пикселе

$$\mathbb{E}_t\mathbf{c} = \int_{t_0}^{t_1}\exp\left(-\int_{t_0}^{t}\sigma(s)\mathrm{d}s\right)\sigma(t)\mathbf{c}(t)\mathrm{d}t$$

Volumetric formulation for NeRF

Оценка вложенного интеграла



Вложенный интеграл вычисляется используя
кусочно-постоянную квадратурную оценку:

Луч разбивается на n сегментов $\delta_i = t_{i+1} - t_i$ точками $\{t_1, t_2, \ldots, t_n\}$.

Предполагается, что на сегментах значение цвета — $\mathbf{c}_i$, и плотности — σ_i, постоянны

Volumetric formulation for NeRF

Оценка вложенного интеграла

Используем эту оценку:

$$\mathbb{E}_t \mathbf{c} = \int_{t_0}^{t_1} T(t)\sigma(t)\mathbf{c}(t)\mathrm{d}t \approx \sum_{i=1}^{n} \int_{t_i}^{t_{i+1}} T(t)\sigma_i \mathbf{c}_i \mathrm{d}t$$

При этом заметим:

$$\forall t \in [t_i, t_{i+1}] \quad T(t) = \underbrace{\exp\left(-\int_{t_1}^{t_i} \sigma_i \mathrm{d}s\right)}_{= T_i} \exp\left(-\int_{t_i}^{t} \sigma_i \mathrm{d}s\right)$$

T_i — не зависит от t

Volumetric formulation for NeRF

Оценка вложенного интеграла

$$\mathbb{E}_t \mathbf{c} = \int_{t_0}^{t_1} T(t)\sigma(t)\mathbf{c}(t)\mathrm{d}t \approx \sum_{i=1}^{n} \int_{t_i}^{t_{i+1}} T(t)\sigma_i \mathbf{c}_i \mathrm{d}t$$

$$\forall t \in [t_i, t_{i+1}] \quad T(t) = \underbrace{\exp\left(-\int_{t_1}^{t_i} \sigma_i \mathrm{d}s\right)}_{= T_i} \exp\left(-\int_{t_i}^{t} \sigma_i \mathrm{d}s\right)$$

$T_i = \exp\left(-\sum_{j=1}^{j-1} \sigma_j \delta_j\right)$ — доля энергии луча не поглощенная на отрезке $[t_1, t_i]$

Volumetric formulation for NeRF

Оценка вложенного интеграла

$$\mathbb{E}_t \mathbf{c} = \int_{t_0}^{t_1} T(t)\sigma(t)\mathbf{c}(t)\mathrm{d}t \approx \sum_{i=1}^{n} \int_{t_i}^{t_{i+1}} T(t)\sigma_i \mathbf{c}_i \mathrm{d}t$$

$$\forall t \in [t_i, t_{i+1}] \quad T(t) = \underbrace{\exp\left(-\int_{t_1}^{t_i} \sigma_i \mathrm{d}s\right)}_{= T_i} \exp\left(-\int_{t_i}^{t} \sigma_i \mathrm{d}s\right)$$

$\exp\left(-\sigma_i(t - t_i)\right)$ — доля поглощенной энергии на отрезке $[t_i, t]$

Volumetric formulation for NeRF

Оценка полного интеграла

$$\mathbb{E}_t\mathbf{c} = \int_{t_0}^{t_1} T(t)\sigma(t)\mathbf{c}(t)\mathrm{d}t \approx \sum_{i=1}^{n}\int_{t_i}^{t_{i+1}} T(t)\sigma_i\mathbf{c}_i\mathrm{d}t$$

$$= \sum_{i=1}^{n} T_i\mathbf{c}_i\sigma_i\int_{t_i}^{t_{i+1}} \exp\left(-\sigma_i(t-t_i)\right)\mathrm{d}t$$

<= Подставим в исходный интеграл вид $T(t)$

$$= \sum_{i=1}^{n} T_i\mathbf{c}_i\sigma_i\frac{\exp\left(-\sigma_i(t_{i+1}-t_i)\right)-1}{-\sigma_i}$$

<= Проинтегрируем

$$= \sum_{i=1}^{n} T_i\mathbf{c}_i\left(1-\exp\left(-\sigma_i\delta_i\right)\right)$$

<= Преобразуем к финальному виду

Volumetric formulation for NeRF

Немного анализа…

$$\ldots = \sum_{i=1}^{n} T_i \mathbf{c}_i \underbrace{\left(1 - \exp\left(-\sigma_i \delta_i\right)\right)}_{\alpha_i}$$

$$\mathbb{E}_t \mathbf{c} = \sum_{i=1}^{n} T_i \mathbf{c}_i \alpha_i$$

α_i — прозрачность!

$$T_i = \prod_{j=1}^{j-1} \left(1 - \alpha_j\right) = \exp\left(-\sum_{j=1}^{j-1} \sigma_j \delta_j\right)$$

Summary: volume rendering integral estimate

Модель рендеринга луча $\mathbf{r}(\mathbf{t}) = \mathbf{o} + t\mathbf{d}$

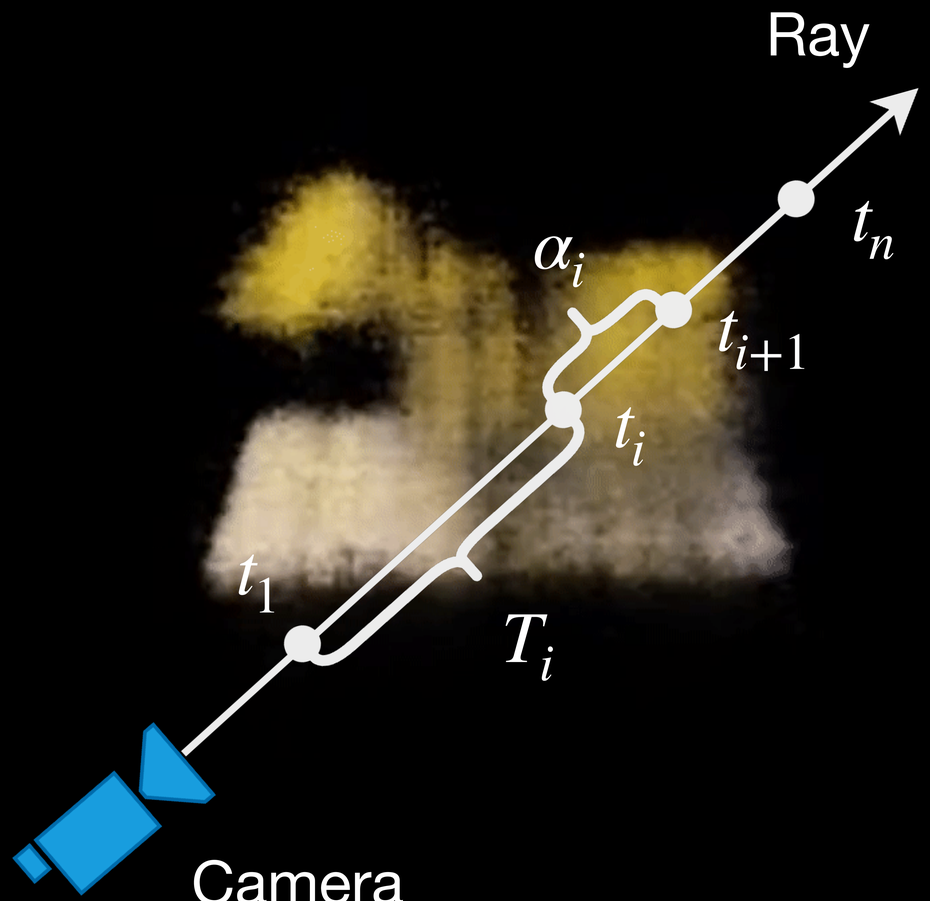
$$\mathbb{E}_t\mathbf{c} = \sum_{i=1}^{n} T_i \mathbf{c}_i \alpha_i$$

T_i — доля заблокированной энергии вдоль луча вплоть до текущего сегмента

$$T_i = \prod_{j=1}^{j-1} \left(1 - \alpha_j\right)$$

α_i — доля пропускаемого света на текущем сегменте

$$\alpha_i = 1 - \exp\left(-\sigma_i \delta_i\right)$$

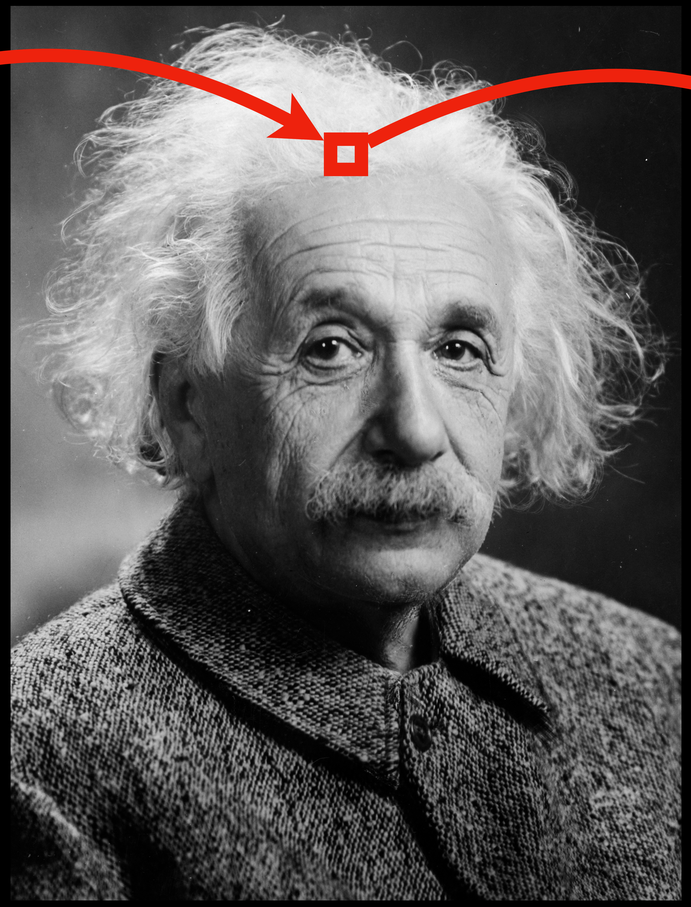


Neural Implicit Representation

Toy 2D Example

Представление изображения

(x, y)

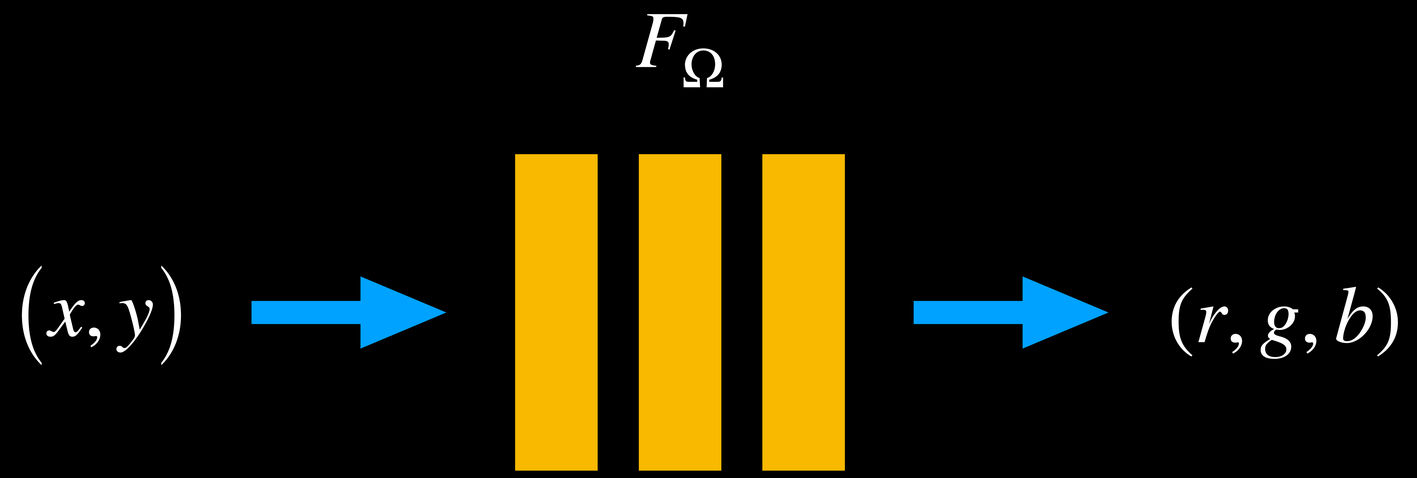


(r, g, b)

Изображение — это сетка пикселей , в которой хранятся цвета

Toy 2D Example

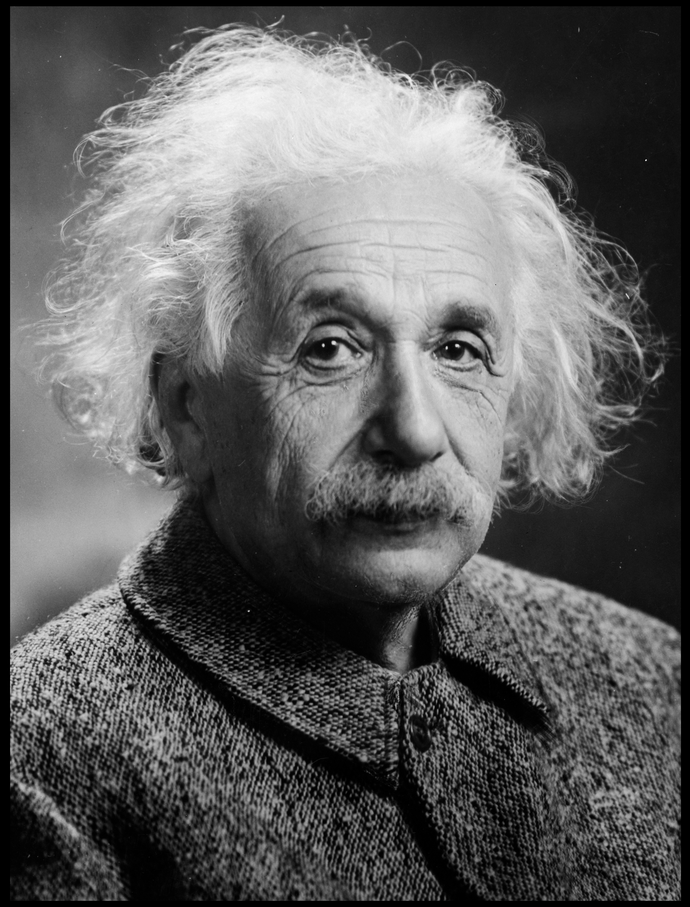
Представление изображения



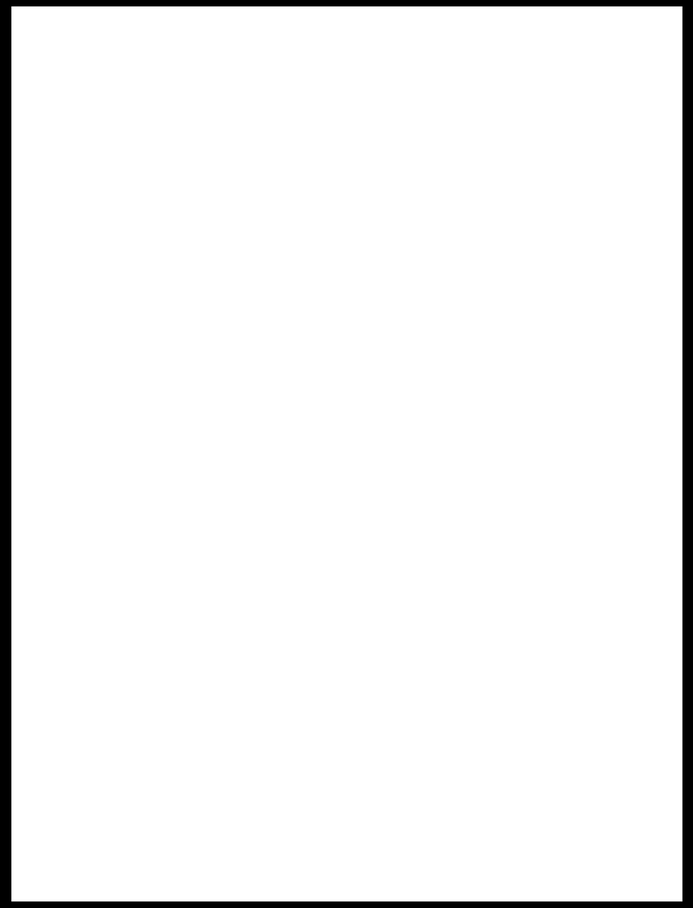
Что будет, если мы попробуем представить
изображение в виде MLP?

Toy 2D Example

Представление изображения. Наивный подход



Original



Naive

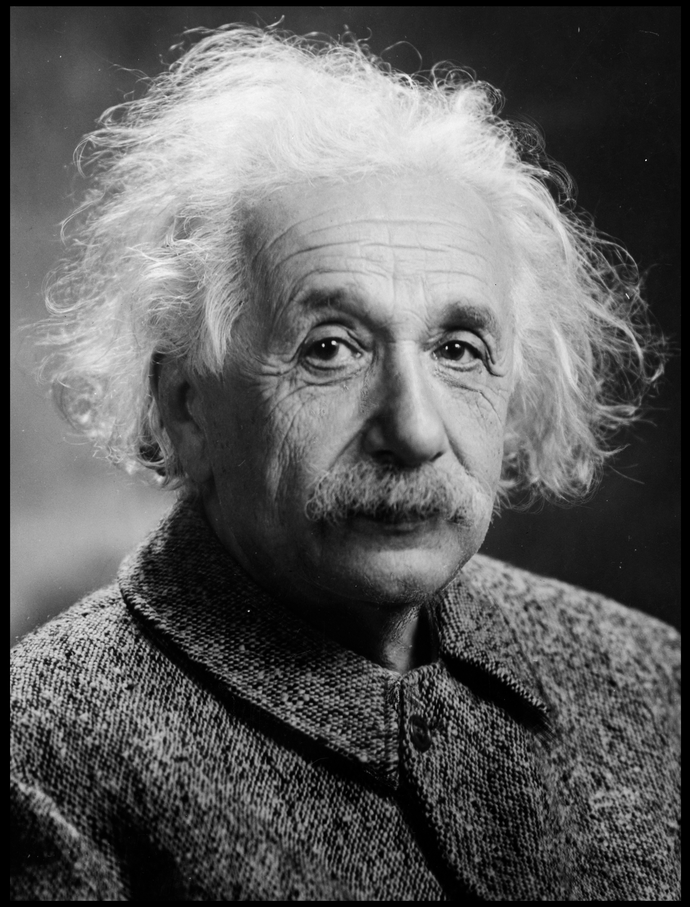
Проблема:
Стандартный покоординатный MLP не может воспроизвести высокочастотные детали изображения

Решение:

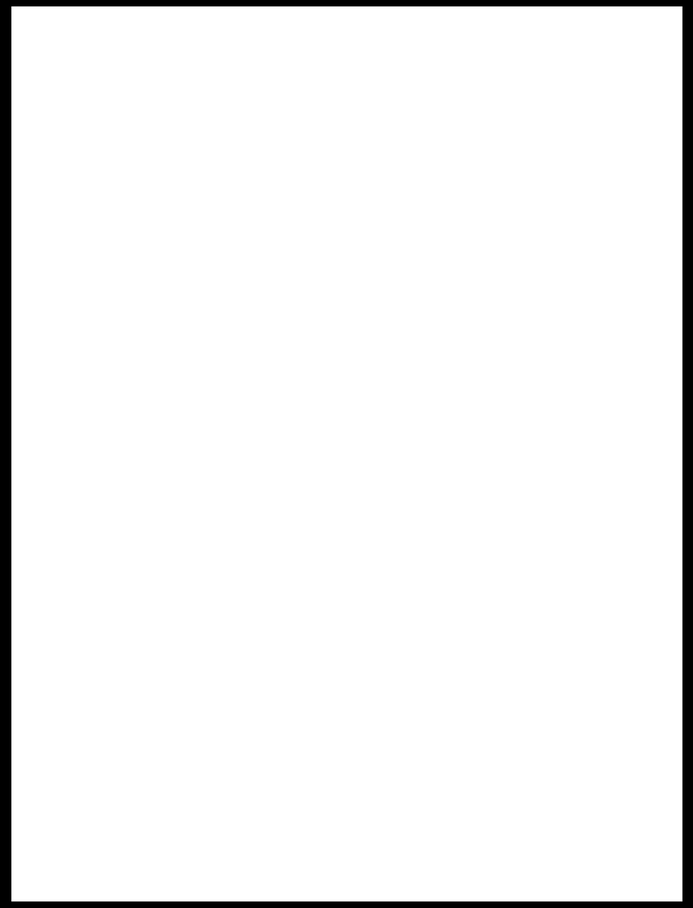
Добавить высокочастотное преобразование координат до подачи их в MLP

Toy 2D Example

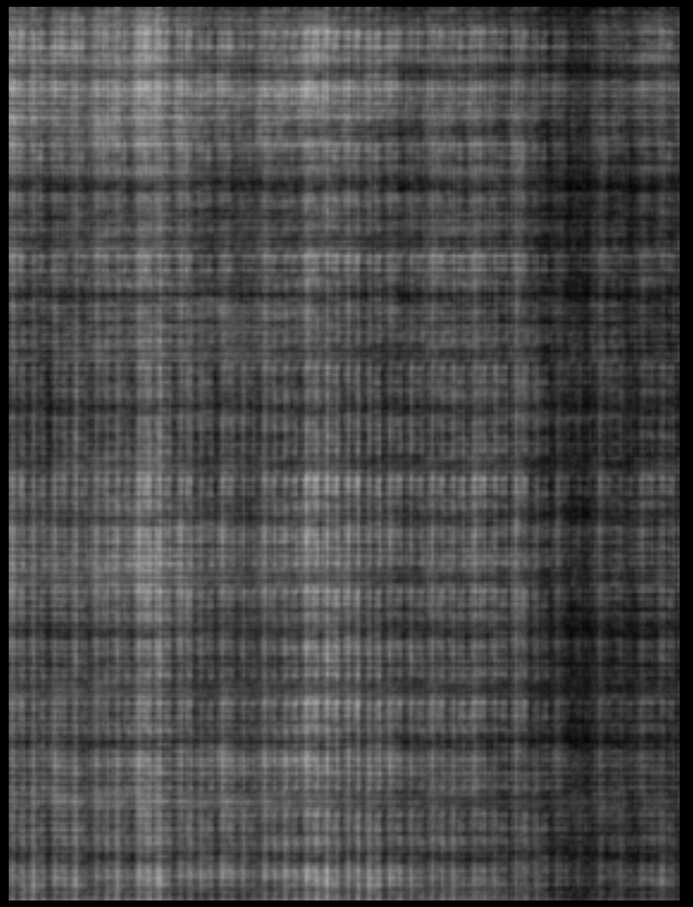
Представление изображения. Positional Encoding



Original



Naive



Positional Enc.

Input coordinate mapping

Positional Encoding

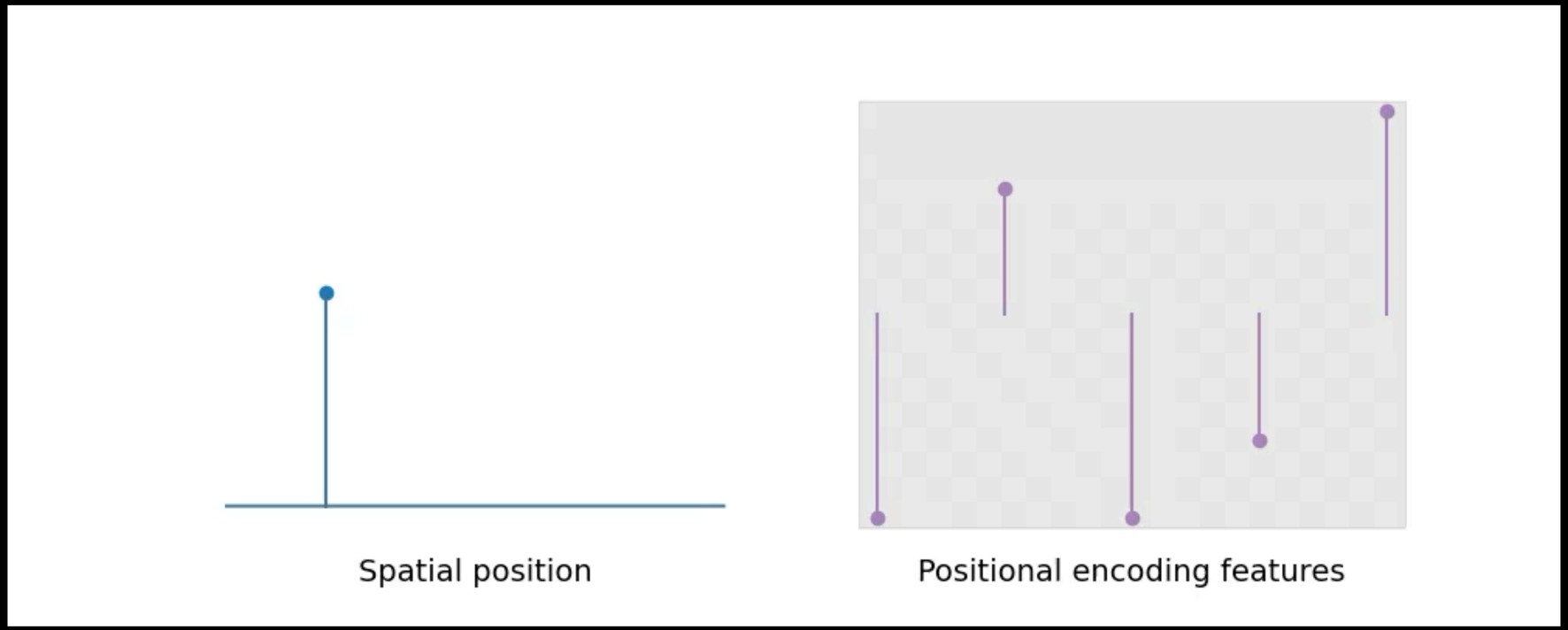
- Общая формула: умножить координатный вектор $\mathbf{x}$ на вытянутую матрицу $\mathbf{B}_{n\times 3}$ для повышения размерности вектора и затем применить $\sin$ и $\cos$ функции:

$$\gamma(x) = \left(\sin(2\pi\mathbf{B}\mathbf{x}), \cos(2\pi\mathbf{B}\mathbf{x})\right)$$

- Такое преобразование переводит координату в подмножество Фурье базисных функций.
- Похожий эффект можно достичь, например, заменой активаций внутри сети на Фурье базисные функции (e.g. SIREN)

Positional Encoding

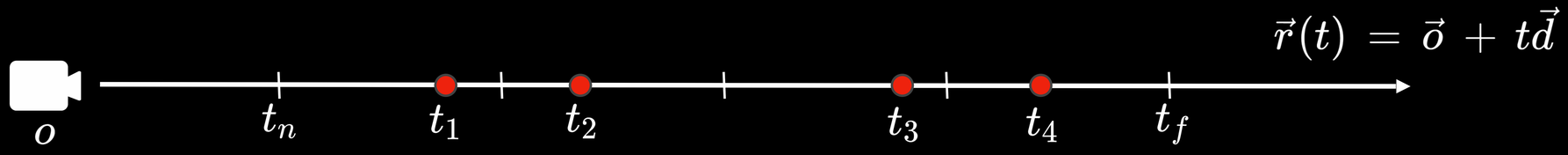
1D example



NeRF = volume rendering +
neural scene representation

NeRF

Сэмплирование



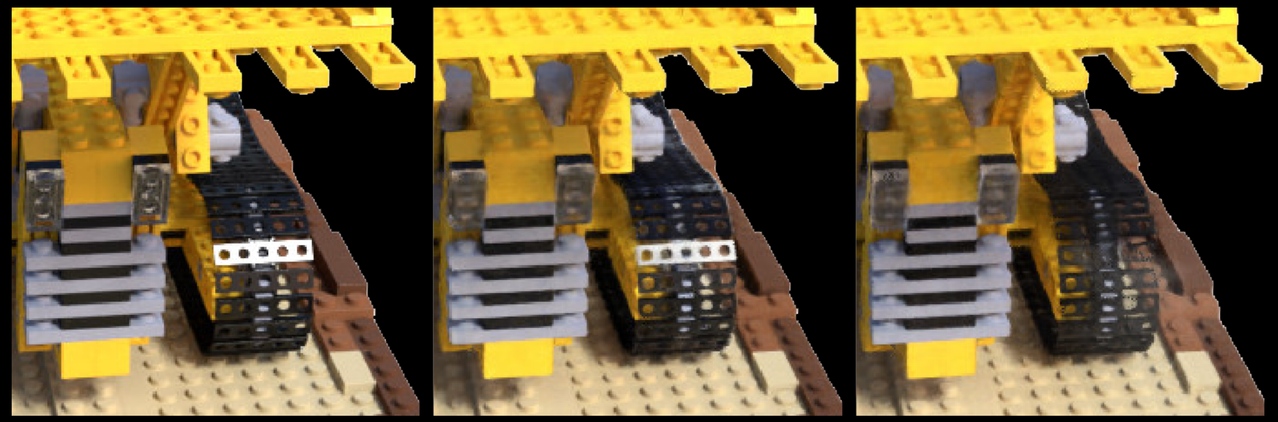
Применяется стратифицированное сэмплирование:

1. Луч, ограниченный near и far плоскостями камеры разбивается на N отрезков
2. Точка внутри отрезка выбирается из равномерного распределения

$$t_i \sim U\left[t_n + \frac{i-1}{N}(t_f - t_n);\ t_n + \frac{i}{N}(t_f - t_n)\right]$$

NeRF

Представление сцены



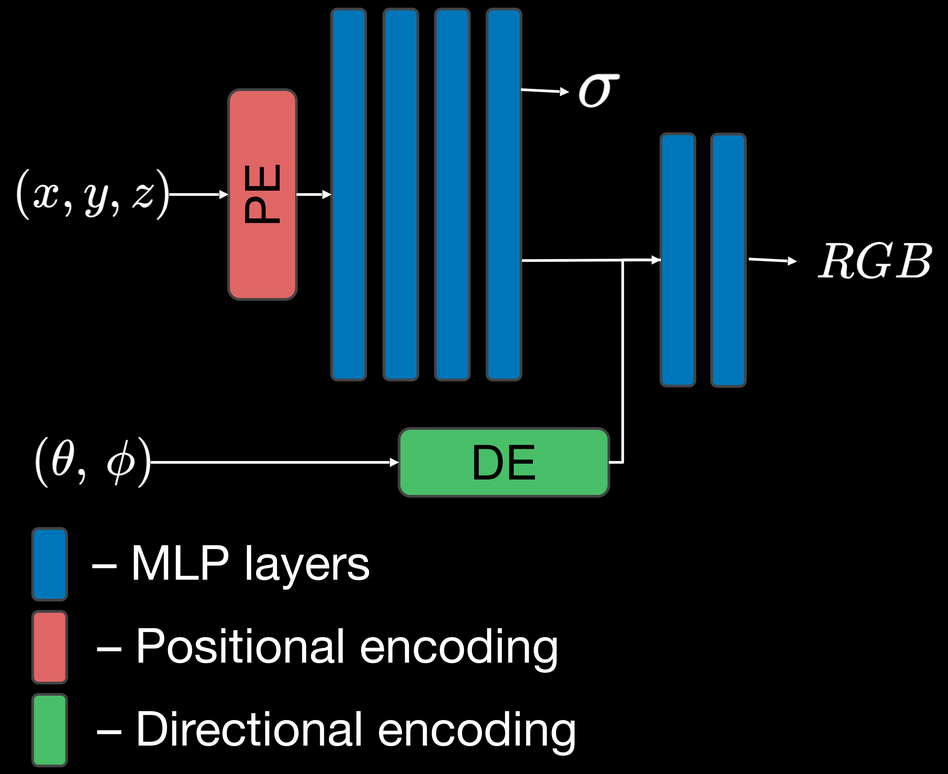
Ground-Truth | With dir. encoding | Positional only

Сцена представляется в виде единого MLP зависящего от (x, y, z, θ, ϕ):

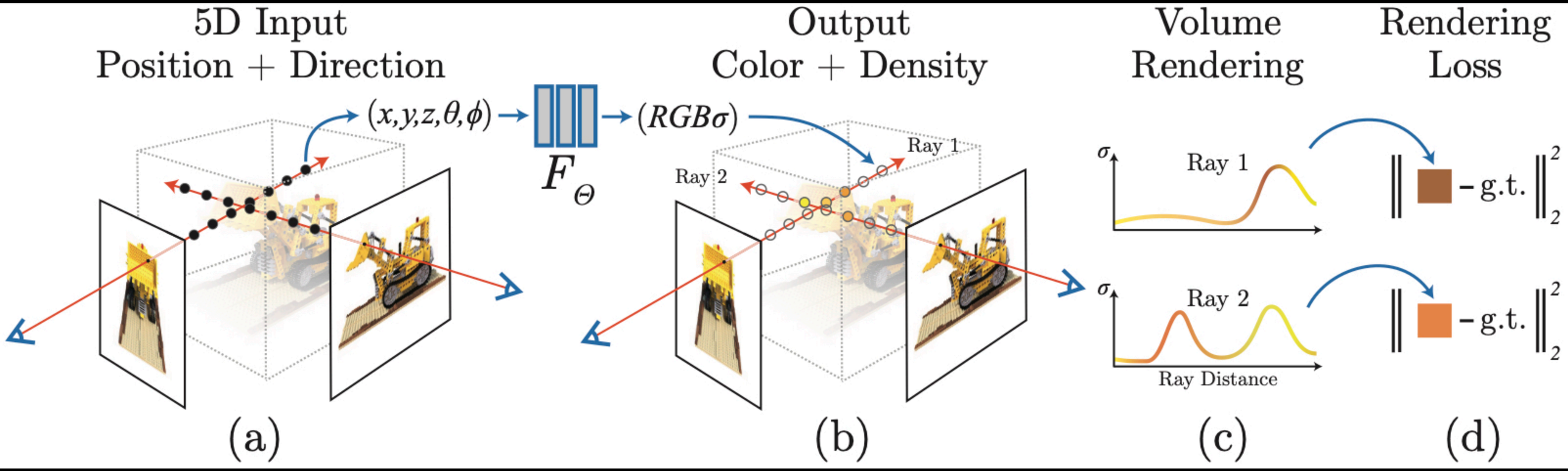
1. Параметры частотно кодируются перед подачей в MLP

2. Зависимость от направления позволяет моделировать сложные эффекты (e.g., specular, transmission)

3. Volume density не зависит от направлений



NeRF Pipeline



Немного примеров





Немного примеров





Немного примеров

View-dependent effects

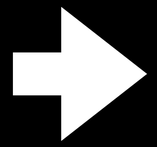


NeRF: depth rendering

Аналогично подходу к оценке цвета луча можно так же оценивать карту глубины

$$\mathbb{E}_t \mathbf{c} = \sum_{i=1}^{n} T_i \mathbf{c}_i \alpha_i$$

Цвет



$$\mathbb{E}_t \mathbf{t} = \sum_{i=1}^{n} T_i t_i \alpha_i$$

Глубина

T_i — доля заблокированной энергии вдоль луча вплоть до текущего сегмента

$$T_i = \prod_{j=1}^{j-1} \left(1 - \alpha_j\right)$$

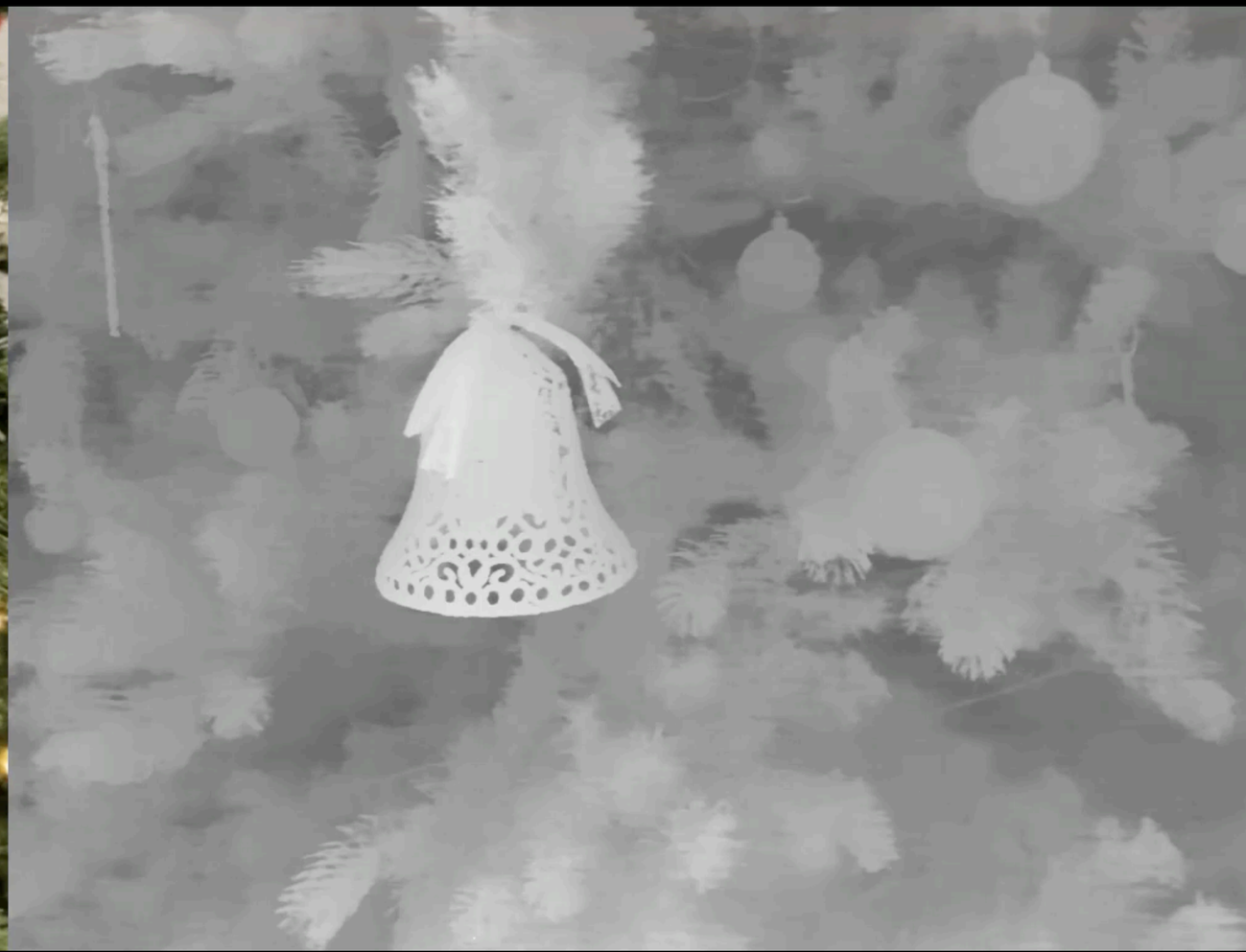
α_i — доля пропускаемого света на текущем сегменте

$$\alpha_i = 1 - \exp\left(-\sigma_i \delta_i\right)$$

Немного примеров

Рендеринг глубины

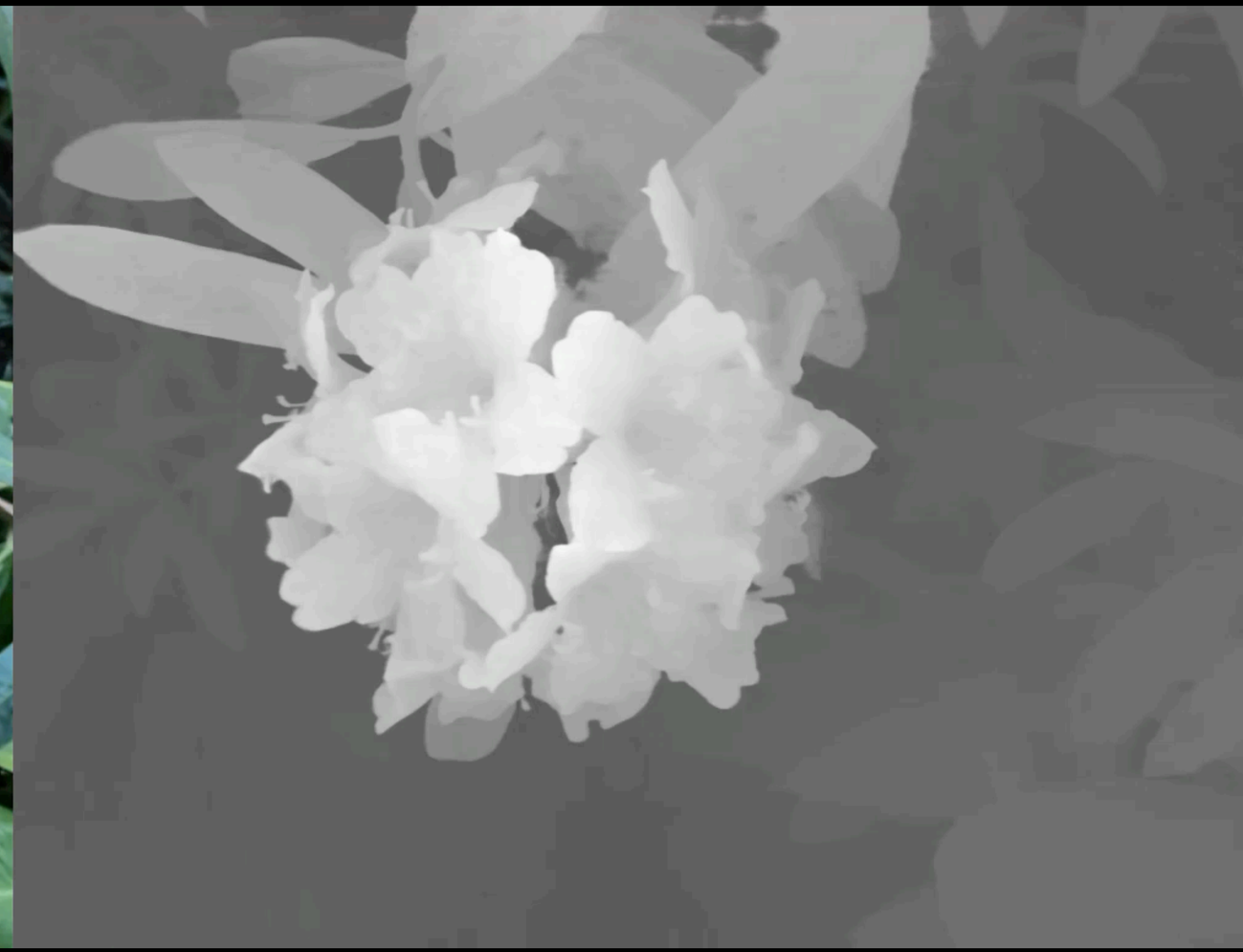




Немного примеров

Рендеринг глубины





Application & Extensions

MipNeRF



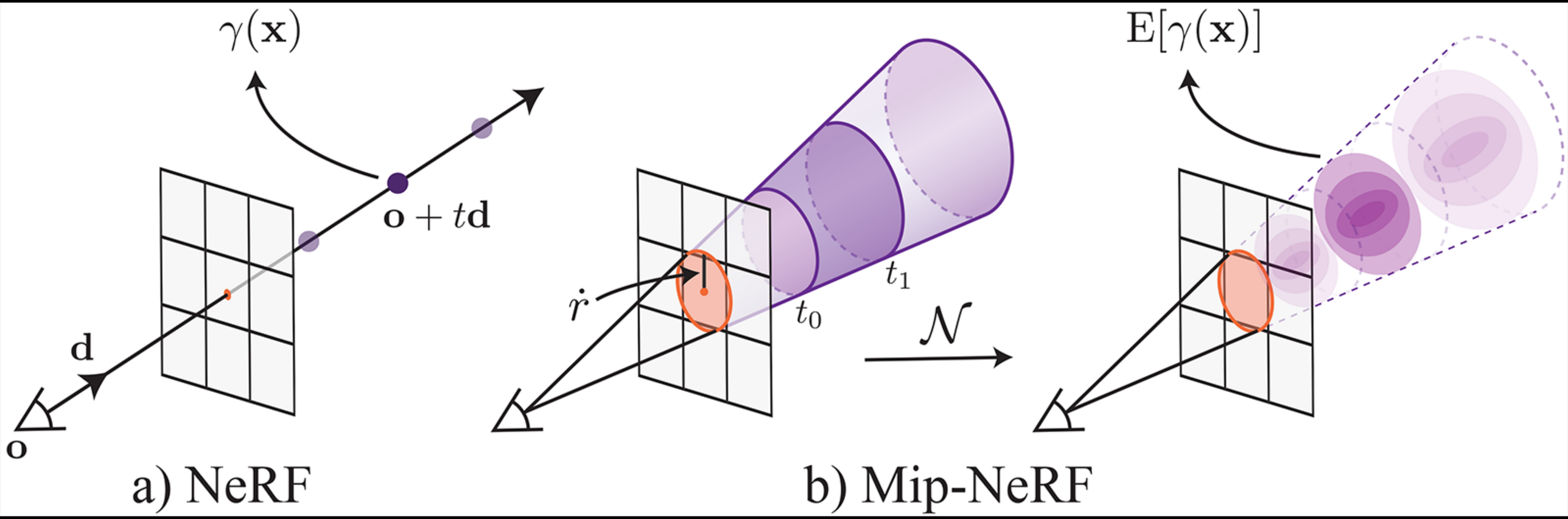


NeRF

MipNeRF

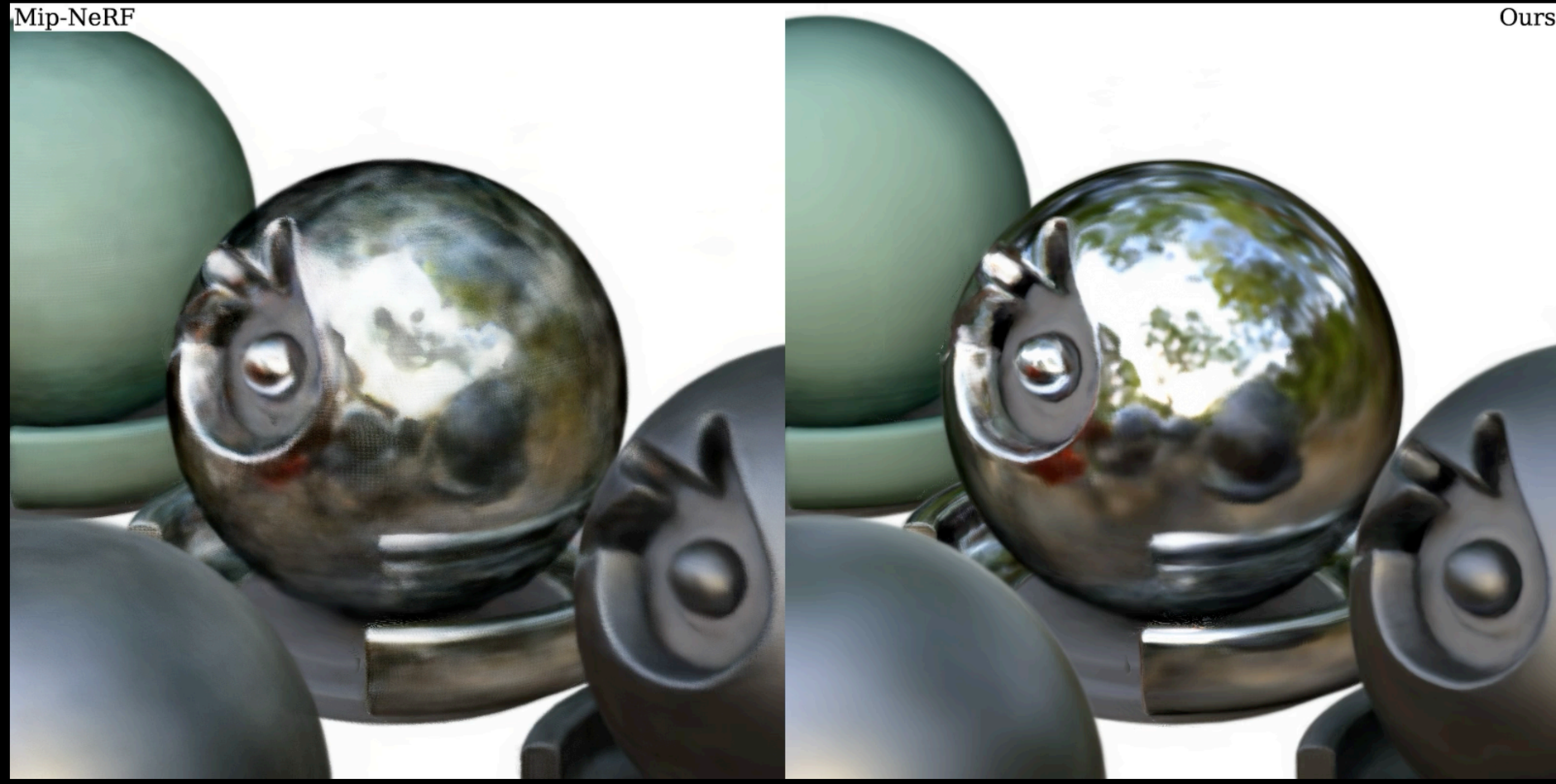
Обычный NeRF не справляется с обучением устойчивого к смене разрешения представления сцены.

MipNeRF



MipNeRF параметризует интервалы интегрирования по лучу используя функции Гаусса — Integrated Positional Encoding

RefNeRF



MipNeRF

RefNeRF

И NeRF, и MipNeRF плохо справляются с моделированием отражений

RefNeRF





RefNeRF частично декомпозирует цвет используя Integrated Directional Encoding, что дает возможность варьировать материалы

Спасибо за внимание!

ждем фидбек

